PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2002-123479
(43)Date of publication of application : 26.04.2002

(51)Int.Cl. G06F 13/10
G06F 3/06
G06F 12/08
G06F 13/00

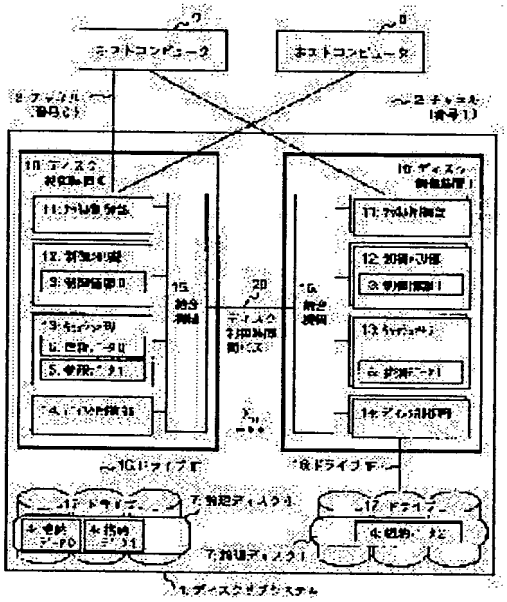
(21)Application number : 2000-316257
(22)Date of filing : 17.10.2000

(71)Applicant : HITACHI LTD
(72)Inventor : KANAI HIROKI
FUJIMOTO KAZUHISA
FUJIBAYASHI AKIRA

(54) DISK CONTROL DEVICE AND METHOD FOR CONTROLLING ITS CACHE

(57)Abstract:
PROBLEM TO BE SOLVED: To perform coincidence control of cache data between devices in a plurality of disk controllers having a cache and to prevent a fault in a specific disk controller from propagating to the disk controllers even if the fault occurs in the disk controller.
SOLUTION: A communicating means between the disk control devices performs data coincidence control. Upon receiving update access from a host, the cache memory of a disk controller for controlling at least a data storage drive performs data update. A cache area is desirably used while being divided between an area for a drive controlled by the disk controller and an area for a drive controlled by the other disk controllers.

図 1



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

* NOTICES *

CLAIMS

[Claim(s)]
[Claim 1] Two or more sets of disk controllers It is disk interface respectively between the means of communications between disk controllers, a disk drive, and a disk controller and a disk drive. Are the disk controller equipped with the above and the aforementioned disk controller is respectively equipped with a cache memory and the control memory which stores the control information of this cache memory. The cache memory with which one disk controller which received the access demand from a host computer was equipped The data to the disk drive connected to the disk controller equipped with this cache memory through the aforementioned disk interface, In addition, it is characterized by making as [hold / the data to the disk drive connected to other at least one disk controller through the aforementioned disk interface are also accessed, and / through the aforementioned means of communications, / data].
[Claim 2] The disk controller according to claim 1 characterized by holding the cache directory which specifies the disk controller which holds the data of an access place on KYASSHI memory, and the cache address which stores the data of this access place for every aforementioned access unit for every access unit for which it opts uniquely from a disk controller number and the disk drive address as control information stored in the control memory section.
[Claim 3] Two or more sets of disk controllers Means of communications between disk controllers. Disk drive. It is disk interface respectively between a disk controller and a disk drive. It is the cache memory control method of the disk controller equipped with the above. The disk controller which the aforementioned disk controller was respectively equipped with the cache memory, and received the access demand from a host computer Process an access demand after the exclusive operation of access data, and a completion report to a host is performed. Access from a host computer is an updating access demand after that. And when disk controllers other than an access receipt disk controller hold these access data to the cache memory, after performing coherence control, it is characterized by canceling the exclusion of these data.
[Claim 4] The disk controller which received the updating access demand from the host computer is the cache memory control method of the disk controller according to claim 3 characterized by storing this disk drive in the cache memory of the disk controller which connected the updating data received from the host through the means of communications between disk controllers through disk interface when it is the updating demand to the drive which this updating access place connected to other disk controllers other than this disk controller through disk interface.

DETAILED DESCRIPTION

[Detailed Description of the Ir.vention]
[0001]
[The technical field to which invention belongs] Many this inventions relate to the control method of the disk controller which constitutes a disk subsystem from two or more sets of disk controllers especially, and the cache memory in a disk controller with respect to the disk subsystem constituted from a magnetic disk unit of a base, and a disk controller which controls these.
[0002]
[Description of the Prior Art] there is a disk controller (it calls Following DKC) which performs much storing and read-out of data to the magnetic disk unit (a following disk drive -- or it is only called a drive) of a base A drive and DKC unite and are named a disk subsystem generically. Such composition of the conventional disk subsystem is shown in drawing 17 .
[0003] In this conventional example, two sets of host computers 0 are connected to two sets of the disk systems 1 through the channel 2, respectively. The logic disk 7 is a storage region which a host computer 0 recognizes. A host computer 0 directs reference of data, and an updating demand to the specific address of the logic disk 7 through a channel 2. There are a fiber channel, SCSI, etc. as a channel 2.
[0004] A disk subsystem 1 consists of two or more sets of DKC10 and drives 17 greatly. Two or more sets of DKC10 and drives 17 are connected with the drive interface (it is called Drive IF below) 16. A fiber channel, SCSI, etc. are used for drive IF 16.
[0005] DKC10 consists of the cache memory section 13 which holds greatly the channel-control section 11 which controls a channel, the disk control section 14 which performs control of a drive, the control memory section 12 which stores the control information 3 of DKC, the reference data 5, and the updating data 6, and a joint mechanism 15 in which each component part is connected further mutually. As for the joint mechanism 15, a bus, a cross coupling network, etc. are used.
[0006] DKC10 performs reference of data, and an update process according to directions of a host computer 0. Such a conventional disk subsystem is indicated by JP,2000-99281,A.
[0007] In a computer environment in recent years, the storage capacity which a user uses is increasing rapidly so that it may be represented by the explosive spread of the Internet. Consequently, increase also of the management cost of the data which increase every day is enhanced, and curtailment of this management cost serves as an important problem. Moreover, it connects for every server conventionally and the storage area network (it calls Following SAN) attracts attention to attain centralization of the disk subsystem distributed as a result. Drawing 18 is the conventional example of the disk subsystem in the SAN environment. Two or more sets of subsystems are connected to a host computer 0 through SAN-SW11. One disk subsystem consists of one set only of a disk controller.
[0008] By change of the environment which surrounds a disk subsystem as explained above, much more storage capacity increase and increase of the number of connection channels are demanded of the disk subsystem.
[0009] From such a background, it is possible to constitute the disk subsystem conventionally

constituted from one set of DKC from two or more sets of DKC(s). Thereby, offer of bigger storage capacity and the number of connection channels is attained as a disk subsystem.
[0010] As common practice which constitutes two or more sets of DKC(s), it is possible to make DKC into a cluster composition. However, the technical problem that sharing of the data between DKC(s) becomes difficult in this case occurs. In order to solve this technical problem, sharing of data is realizable by using the connecting means between DKC(s) and enabling access of data mutually between DKC(s).
[0011]
[Problem(s) to be Solved by the Invention] However, in the disk subsystem which consists of two or more sets of DKC(s) in which a data access is possible mutually, the data coincidence problem between the cache memories between each DKC is important. Generally this is called coherence control.
[0012] Especially, since the data of a drive are the last storage section of user data in the case of a disk subsystem, it is important to guarantee this data, for example, even when an obstacle is occurred and downed to one set of DKC in a disk subsystem, an obstacle must not spread to other DKC(s). However, since access to this data becomes impossible when being left behind to the cache memory of DKC which the obstacle generated, where the data of a drive linked to DKC in which other normal operation is possible are updated, normally DKC which manages the drive which stores this data, and this drive will not be involved, but DETAROSUTO will arise. That is, the obstacle of one set of DKC in a subsystem will spread to other DKC(s) in the same subsystem, and poses a problem.
[0013]
[Means for Solving the Problem] In addition to the data to the drive linked to the disk controller which received the access demand from a host, each disk controller was equipped with the cache memory holding the data to the drive linked to other disk controllers other than the disk controller which received the access demand from a host through the means of communications between these disk controllers, and the control memory which stores the control information of this cache memory.
[0014] Furthermore, this disk controller prepared the cache management table holding the cache directory which can specify the disk controller currently held on a cache with reference to the data of an access place from the address of a disk controller and a drive for every access unit which can be determined as a meaning as control information for controlling the cache which it had in the disk controller, and the cache address which stores the data of this access place.
[0015] After carrying out coherence control, the exclusion of these data was made for the disk controller which received the access demand from a host computer to cancel, when access from a host computer is an updating access demand and disk controllers other than an access receipt disk controller hold these access data to the cache memory, after performing the exclusive operation of access data at the beginning of processing, processing an access demand and carrying out a completion report to a host after that.
[0016] The disk controller which received the updating access demand from the host computer stored the updating data received from the host computer through the means of communications between disk controllers in the cache memory of the disk controller which connected this drive, when this updating access place was the updating demand to the drive linked to other disk controllers other than this disk controller.
[0017] The coherence control method updated the data which hold the data currently held into the cache of other disk controllers to nullification or the cache memory of other disk controllers.

[0018] The disk controller which received the host computer refernce access demand It judges whether it is held at the cache in the disk controller with which access data received the access demand with reference to the directory of the cache management table of the disk controller which connects the drive of introduction and an access place. When these data are held, with reference to this cache, these data are immediately transmitted to a host computer. On the other hand, when these access data are not held at the cache in the disk controller which received the access demand It judges whether it is held at the cache memory of the disk

controller to which access data connect the drive of this access place with reference to the directory of the cache management table of the disk controller which connects the drive of an access place. When these data are held, with reference to this cache memory, these data are immediately transmitted to the cache and host computer in the disk controller which received the access demand. On the other hand, when not held at the cache memory of the disk controller which connects the drive of this access place It was made to transmit to the cache memory of the disk controller which connects the drive of this access place for these data, the cache memory in the disk controller which received this access demand, and a host computer from a drive.

[0019] When a cache field released, it stores in the drive which connects to this disk controller the updating data held into this cache, and these data of the cache memory of another disk controller which holds these data within a disk subsystem were cancelled further.

[0020] The cache memory with which each disk controller was equipped transmitted data to the host from the cache of the disk controller of a demand place, or the drive, when the access demand from a host computer was reference by holding only the data of a drive linked to this disk controller, or when the access demand from a host computer was updating, it transmitted data to the cache of the disk controller of a demand place.

[0021] The storing field of a cache memory divides a field into the storing field of the data to the drive linked to the disk controller which received access, and the storing field of the data to the drive linked to other disk controllers in a subsystem, and managed it.

[0022] On the cache memory, it doubled or multiplexed and the data to the drive linked to the disk controller which received access stored data, and on the other hand, the data to the drive linked to other disk controllers in a subsystem were stored without multiplexing on a cache.

[0023] The cache with which a disk controller is equipped consisted of volatilization caches which store the data to the drive linked to the non-volatilized cache which stores the data to the drive linked to the disk controller which received access, and other disk controllers in a subsystem.

[0024] When an obstacle occurred in a certain disk controller in a subsystem, the data of a drive linked to this obstacle generating disk controller currently held to the cache memory of a normal disk controller were cancelled.

[0025] It was made for the means of communications between disk controllers to be a host computer, a part of connectable channel, and a switch that connects these channels.

[0026] As control information for controlling the cache which it had in the disk controller A channel, a disk controller, and the access log table holding the access frequency information for every logic disk are prepared. It judges whether the logic disk of the channel with the highest access frequency among the channels which receive access to a certain logic disk, and this access place is connected to the same disk controller. When not the same, the channel with this highest access frequency rearranged this logic disk on the drive of the connected disk controller. Moreover, when the same, the host computer which uses other channels which access this logic disk used the channel of the disk controller which connects this logic disk.

[0027]

[Embodiments of the Invention] Hereafter, the detail of invention is explained using a drawing. The disk equipment concerning this invention is explained using introduction, drawing 1 , and drawing 2 . Drawing 1 is an example of the block diagram showing the outline of the disk controller concerning this invention. A disk subsystem 1 is connected to a host computer 0 through two or more channels 2. this example -- a disk subsystem 1 -- two or more sets of DKC(s)10 -- constituting -- every -- DKC10 -- every -- with reference to the storing data 4 stored in the drive 17 connected to DKC10 through the path 20 between disk controllers of exclusive use at other DKC(s), the feature is in the place which can be updated Hereafter, it explains in detail.

[0028] The disk subsystem 1 shown in drawing 1 consists of drives 17 with two or more sets of DKC(s)10 greatly. Although two details of DKC10 are shown in the detail, each composition of each of DKC10 of Xn base is the same. DKC10 consists of the cache memory section 13 which holds greatly the channel-control section 11 which controls a channel, the disk control section

14 which performs control of a drive, the control memory section 12 which stores the control information 3 of DKC, the reference data 5, and the updating data 6, and a joint mechanism 15 in which each component part is connected further mutually. Although illustration has not been carried out, the channel-control section 11 and the disk control section 14 are equipped with the processor for control, and a processing program operates on a processor.

[0029] Drawing 2 is the block diagram having shown an example of the control information stored in the control memory 12 in drawing 1 . At this example, control information 3 consists of a cache management table 31 and an equipment configuration managed table greatly.

[0030] First, the cache management table 31 is explained in full detail. The cache management table 31 holds directory information and cache address information. In this example, on explanation, although directory information and cache address information are described as an individual table, these are good also as the same table. Directory information shows the relation holding the data of the access place drive address and the address of a host of a cache. Specifically, as a host's access place address, it has the disk controller number and the drive address of an access place, and each cache in a subsystem is the directory in which it is shown for every cache whether the data to the address are held further. By this example, the cache directory is prepared for every DKC, and when a cache directory is 0 again about the cache of DKC holding data when a cache directory is 1, it shows that the cache of DKC does not hold data. Therefore, it is shown that the data of the disk controller number 0 and the drive address 0 are stored in the cache of DKC0. Moreover, it is shown that the data of the disk controller number 0 and the drive address 1 are stored in both the cache of DKC0 and the cache of DKC1.

[0031] Next, although it is cache address information, this shows the relation of a host's access place drive address and the cache address holding the data of the address. Like directory information, as the access place address of a host computer, it has the disk controller number and the drive address of an access place, and the cache address holding the data to the address is shown. Cache address information may hold only the address over the cache in DKC.

[0032] Next, an equipment configuration managed table is explained in full detail. An equipment configuration managed table shows the relation between a channel number and a logic disk with an identifiable host computer, and the actual data storage point drive in DKC. An example shows that the logic disk 0 of a channel number 1 is assigned to the drive number 0 of the disk controller number 0. Moreover, it is shown that the logic disk 1 of a channel number 1 is assigned to the drive number 1 of the disk controller number 1.

[0033] It is identifiable in the disk controller number and drive number of an access demand place of a host by referring to the equipment configuration managed table 32, if the control information 3 explained above is used, and it is still more possible to understand the cache address which holds the data of an access place by illuminating the cache address information of the cache memory which is identifiable and corresponds further the cache memory number which holds the data of an access place by referring to the directory information on the cache management table 31 3.

[0034] In this example, although the equipment configuration managed table 32 is stored in the control memory section 12, you may store the storing place of this equipment configuration managed table 32 on the local memory of the control processor with which the channel-control section 11 and the disk control section 14 were equipped.

[0035] Next, operation of DKC at the time of receiving the access demand from a host computer using the flow chart shown in drawing 10 from drawing 3 and the cache memory control method are explained.

[0036] The outline of processing is shown using introduction and drawing 3 . Drawing 3 is the flow chart having shown the flow of the whole processing of DKC. The processing shown with this flow chart is realizable as a processing program on the processor in DKC. DKC which received the access demand from a host computer performs processing of a receiving command, and the exclusive operation of access data first (Step 1). carrying out command analysis according to the protocol of a connection channel -- an access demand -- a lead command -- namely, It is discriminable whether they are whether it is a reference demand and a light

command, i.e., an updating demand. Furthermore, an exclusive operation is performed so that access place data may not be updated and referred by other access processings. What is necessary is just to perform an exclusive operation by the common practice by the lock etc. Next, DKC performs reference or an update process according to a command (Step 2). Next, a completion report to a host is performed after a processing end (Step 3). A receipt command is judged (Step 4). When it is not a light command, it progresses to the below-mentioned step 7. When it is a light command next, it judges whether other DKC(s) in a subsystem can hold the data of the address concerned which the host accessed to a cache memory (Step 5). When other DKC(s) cannot hold to a cache memory, it progresses to the below-mentioned step 7, and processing is ended. When other DKC(s) can hold into a cache, coherence processing of updating data is performed (Step 6). Finally, the exclusion of the data of an access place is canceled (Step 7). In this example, when other DKC(s) hold the old data on a cache at the time of light command processing, the feature is in the place which performs coherence processing of data. The detail of command processing or coherence processing is mentioned later, respectively.

[0037] Drawing 4 is the flow chart showing an example of the processing at the time of light command receipt. If a command is received, from a receiving command, a command and the access place address will be analyzed and it will recognize that it is light access (Step 1). The access place address is referring to an equipment configuration managed table, and can discriminate the disk controller number and drive number of an access demand place. Next, a cache hit mistake judging is performed to the cache memory concerned of DKC discriminated at Step 1 (Step 2). It is identifiable in whether access place data are held by referring to the directory information on a cache management table at the cache. The DKC concerned carries out the disk control-section pair of the case of the cache mistake which is not held into a cache, and it performs the transfer request to a cache memory from the drive of the data concerned (Step 6). Usually, this processing is called staging processing. In this case, light processing will be interrupted till a transfer end (Step 7), and light processing will be again continued after a staging end. Moreover, although what is necessary is to manage the cache address of the destination and just to acquire it by common practice, such as an empty list of caches, it is necessary to register it by updating a cache management table for the destination address. When staging processing is completed at the case of a hit judging, or Step 7 by Step 3, the data concerned are updated to the cache memory concerned of DKC (Step 4). The completion report of light processing is performed to a host computer after an updating end (Step 5). In this example, the feature is in the place which can access all the cache memories of DKC in a subsystem by referring to the cache directory and the cache address of a cache management table.

[0038] Drawing 5 is the flow chart showing an example of the coherence control of a cache memory performed following the light processing at the time of light command receipt. In this example, the feature is in the place which cancels the data of other cache memories. It judges whether there is any cache memory of other DKC(s) which hold the old data of the updating demand address by referring to the directory information on a cache management table (Step 1). It is an end when there is no cache memory of other DKC(s) holding the old data. Directory information is updated when there is a cache memory of other DKC(s) holding the old data. For example, what is necessary is just to write in 0 as directory information in the example, since 1 shows the maintenance state. Furthermore, it is necessary to release the old data-hold field of the cache memory concerned. The old data-hold address of the cache memory concerned can be recognized by referring to the cache address information concerned of DKC. This cache address is deleted from this cache address information, and empty list ** of the above-mentioned cache memory is good in this cache memory field. By the above, the cache memory of other DKC(s) holding the copy of data is cancelled (Step 2).

[0039] Drawing 6 is the flow chart showing other examples of the coherence control of a cache memory performed following the light processing at the time of light command receipt. In this example, the feature is in the place which updates the data of other cache memories. It judges whether there is any cache memory of other DKC(s) which hold the old data of the updating demand address by referring to the directory information on a cache management table (Step 1).

It is an end when there is no cache memory of other DKC(s) holding the old data. Data are updated when there is a cache memory of other DKC(s) holding the old data. The old data-hold address of the cache concerned can be recognized by referring to the cache address information concerned of DKC. What is necessary is just to write in updating data to this cache address. By the above, the cache of other DKC(s) holding the copy of data concerned is updated (Step 2).

[0040] Drawing 7 and drawing 8 are the flow charts showing an example of the processing at the time of lead command receipt. The receipt command is expressed as a receiving command drawing. this example shows the case where the maintenance to the cache memory of the data which other DKC(s) manage between DKC(s) is possible. If a command is received, from a receiving command, a command and the access place address will be analyzed and it will recognize that it is lead access (Step 1). The access place address is referring to an equipment configuration managed table, and can discriminate the disk controller number and drive number of an access demand place. Next, a cache hit mistake judging is performed to the cache memory concerned of DKC discriminated at Step 1 (Step 2). It is identifiable in whether access place data are held by referring to the directory information on a cache management table at the cache memory. Whether it holds to the cache memory of the command receipt DKC, the data judging (Step 3) and holding to the cache memory of the command receipt DKC, and when concerned are immediately referred to to the cache memory of this command receipt DKC (Step 4). Furthermore, the data concerned are transmitted to a channel (Step 5). On the other hand, in a cache mistake, it judges whether it holds to the cache memory of DKC which connects the drive of an access place at Step 3 (Step 6). When holding to the cache memory of DKC which connects the drive of an access place, data are transmitted to the cache memory of the command receipt DKC from the cache memory of DKC which connects the drive of this access place. Under the present circumstances, although what is necessary is just to acquire the data storage point address of the cache memory of this command receipt DKC from the empty list of the above-mentioned cache memories etc., it needs to write this address in the cache address information of the cache management table of this command receipt DKC. Furthermore, the directory information on the cache management table of DKC which connects the drive of an access place is updated so that having copied data to the cache of this command receipt DKC may be shown (Step 7). It progresses to Step 4 of point ** after the waiting for a transfer end (Step 8). On the other hand, when it becomes a cache mistake at Step 6, it is necessary to read data from a drive. Usually, this processing is called staging processing (Step 9), and this step is the same as Step 6 of drawing 4 . It progresses to Step 4 of point ** after the waiting for a transfer end (Step 10).

[0041] Drawing 9 is the flow chart showing an example of the release method of a cache field when the maintenance to the cache memory of data is possible between DKC(s). When a free area is lost into a cache, according to a predetermined algorithm, it is necessary to release the field of a cache. There is LRU rule as a general algorithm. By this method, the field first released according to a predetermined algorithm is determined. Then, it judges whether it is data of a drive which the data stored in a release field now connect to self-DKC (Step 1). This judgment can be performed by referring to the cache address information of a cache management table. When it is not data of a drive linked to self-DKC, the directory information on the cache management table of DKC which connects the data storage point drive concerned and is managed is updated, and it is made for the cache memory of self-DKC not to hold data (Step 5). On the other hand, at Step 1, when it judges with it being data of a drive linked to self-DKC, the writing to the drive of applicable data is requested to a disk control section (Step 2). This processing is usually called DESUTEJI processing. After waiting for a write-in end (Step 3), according to the directory information on a cache management table, other DKC(s) holding this copy of data carry out a cache pair, and the data concerned are cancelled (Step 4).

[0042] Drawing 10 is the flow chart showing an example of the processing at the time of lead command receipt. this example shows the case where the maintenance to the cache memory of the data which other DKC(s) manage between DKC(s) is impossible. If a command is received, from a receiving command, a command and the access place address will be analyzed and it will recognize that it is lead access (Step 1). The access place address is referring to an equipment

configuration managed table, and can discriminate the disk controller number and drive number of an access demand place. Next, a cache hit mistake judging is performed to the cache memory concerned of DKC discriminated at Step 1 (Step 2). It is identifiable in whether access place data are held by referring to the directory information on a cache management table at the cache. The data concerned are referred to to the cache of whether it holds to the cache memory of DKC which connects the drive of an access place, and DKC which connects the drive of this access place immediately when holding into the cache of DKC which judges (Step 3) and connects the drive of an access place (Step 4). Furthermore, the data concerned are transmitted to a channel (Step 5). On the other hand, in a cache mistake, it is necessary to read data from a drive at Step 3. Usually, this processing is called staging processing (Step 6), and this step is the same as Step 6 of drawing 4 . It progresses to Step 4 of point ** after the waiting for a transfer end (Step 7).

[0043] Next, other examples with the desirable management method of the cache memory section 13 are explained using drawing 11 . In this example, the feature is in the place which divided the field of the cache memory section 13 into the data storage field for other DKC(s), and the data storage field for self-DKC. a part of [consequently, / the doubleness of data mentioned later, multiplexing, or a part of cache] -- realization becomes easy and possible about un-volatilizing-ization at a low cost In order to divide a field, the list which manages the free area of a cache memory is required for every field.

[0044] In this example, only the data storage field for self-DKC is doubled. The updating data from a host are held to the cache memory of DKC which the storing place drive of these data connects, i.e., the data storage field for self-DKC. Therefore, reliability can be improved by doubling this data storage field for self-DKC. Moreover, as compared with the case where the cache memory section 13 whole is doubled by doubling only this data storage field for self-DKC, reliability is securable by the low cost.

[0045] Next, other examples with the desirable management method of the cache memory section 13 are explained using drawing 12 . The feature is in this example to constitute the cache memory section 13 from a volatilization cache field 131 which stores the data for other DKC(s), and a non-volatilized cache field 132 for self-DKC which carries out data storage.

[0046] Next, other examples of a disk controller are shown using drawing 15 and drawing 16. Drawing 15 is the block diagram having shown other examples of the control information stored in the control memory 12 in drawing 1 . In this example, the feature is in the place equipped with the access log table 33 showing the statistics of the number of times of access from a host other than a cache management table and an equipment configuration managed table as control information. The access log table 33 shows the number of times of access for every channel number, logic disk number, and disk controller number. In this example, it divides into the number of times of a lead, and the number of times of a light, and holds. The number of times of access should just be made to carry out renewal of an increment of the number of times in accordance with the time of each processing program accessing control information at the time of the receipt command analysis from a host, or a cache hit mistake judging. A host's access property can be recognized in analyzing each access log table 33 of DKC in a subsystem. In this example, access to the logic disk number of No. 0 is accessed from channel numbers 0, 1, and 3, and access from a channel 1 is understood that the data transfer through the path 20 between disk controllers is required.

[0047] Next, the example of the discernment method of concrete access frequency is explained using drawing 16 . Drawing 16 is the flow chart of processing of the access character-recognition method which the processing program on the processor in DKC performs. Although it assumes that the processing program on the processor in DKC performs in this example, the processing program on the administrative processor besides DKC may perform. According to execution or the directions from a host, what is necessary is just made to perform access character-recognition processing with a timer periodically. About access to the logic disk [table / access log / 33 / introduction and / of each disk controller] of a specific disk controller, the number of times of access for every channel number is extracted, and the number of times of access for every channel is compared (Step 1). Next, the number of times of access

judges whether the DKC number to which the greatest channel was connected is the same as the DKC number to which the logic disk for analysis was connected among the number of times of access extracted and compared (Step 2). By this judgment, it judges whether the channel with the high access frequency to a logic disk is connected to the same DKC as a logic disk. when the same at the judgment of Step 2, the greatest channel and the greatest logic disk have the same number of times of access -- since it connects with DKC, the channel of DKC which has connected the logic disk is used for the channel which has the need of using the path 20 between disk controllers, among other channels which are unnecessary as for relocation of a logic disk, and have accessed this logic disk (Step 3) What is necessary is just to perform directions to the processor for configuration managements of DKC, or a host computer. on the other hand, when not the same at the judgment of Step 2, the greatest channel and the greatest logic disk have the same number of times of access -- since it does not connect with DKC, it is good to connect this logic disk with this channel at the same DKC In this example, the number of times of access directs that the logic disk concerned rearranges to the drive of DKC which connects the greatest channel (Step 4). Since the operating frequency of the path between disk controllers can be made low and access from a host can be carried out to access to the logic disk of DKC which received access by repeating Step 4 about all the logic disks in a subsystem, and performing it from Step 1, even if the band of the path between disk controllers is low, it changes so that a performance can be maintained.

[0048]

[Effect of the Invention] Each disk controller is added to the data to the drive linked to the disk controller which received the access demand from a host. The means of communications between these disk controllers is minded. Since it had the cache memory holding the data to the drive linked to other disk controllers other than the disk controller which received the access demand from a host, and the control memory which stores the control information of this cache memory Between the caches which it had in each disk controller, it becomes sharable [data] and the improvement in a performance can be carried out.

[0049] Furthermore, since this disk controller prepared the cache management table holding the cache directory which can specify the disk controller currently held on a cache memory with reference to the data of an access place from the address of a disk controller and a drive for every access unit which can be determined as a meaning as control information for controlling the cache memory which it had in the disk controller, and the cache address which stores the data of this access place, the coherence control of a cache memory of it is attained.

[0050] The disk controller which received the access demand from a host computer After performing the exclusive operation of access data at the beginning of processing, processing an access demand and performing a completion report to a host after that Access from a host computer is an updating access demand. further When disk controllers other than an access receipt disk controller hold these access data into the cache Since the exclusion of these data was canceled after performing coherence control, coherence control can be realized without increasing the response time of a host computer.

[0051] The disk controller which received the updating access demand from the host computer When this updating access place is the updating demand to the drive linked to other disk controllers other than this disk controller Since the updating data received from the host through the means of communications between disk controllers were stored in the cache memory of the disk controller which connected this drive Even when an obstacle occurs in a certain disk controller in a disk subsystem, the data of other disk controllers can prevent propagation of an obstacle, without ROSUTO.

[0052] Since the coherence control method cancelled the data currently held to the cache memory of other disk controllers, even when the transfer band of the connecting means between disk controllers is low, the coherence control of it is attained.

[0053] Moreover, since the data currently held to the cache memory of other disk controllers as the another coherence control method were updated, the hit ratio of a cache memory improves more and a performance is improved.

[0054] The disk controller which received the host computer refernce access demand judges

whether it is held at the cache memory in the disk controller with which access data received the access demand first with reference to the directory of the cache management table of the disk controller which connects the drive of an access place. When these data are held, with reference to this cache memory, these data are immediately transmitted to a host computer. On the other hand, when these access data are not held at the cache memory in the disk controller which received the access demand, it judges whether it is held at the cache memory of the disk controller to which access data connect the drive of this access place with reference to the directory of the cache management table of the disk controller which connects the drive of an access place. When these data are held there, with reference to this cache memory, these data are immediately transmitted to the cache memory and host computer in the disk controller which received the access demand. When not held at the cache memory of the disk controller which connects the drive of this access place, it was made to transmit to the cache memory of the disk controller which connects the drive of this access place for these data, the cache memory in the disk controller which received this access demand, and a host computer from a drive on the other hand. Therefore, even if it is data of a drive linked to disk controllers other than the disk controller which received the access demand, it can refer to, and when these access data are held at the cache memory, compared with the case where a drive is accessed, data can be further transmitted to a host computer by the short response time.
 [0055] When releasing a cache field, since it stores in the drive which connects to this disk controller the updating data held to this cache memory and these data of the cache of another disk controller which holds these data within a disk subsystem further were cancelled, a cache can be used efficiently.
[0056] The cache memory with which each disk controller was equipped was made to hold only the data of a drive linked to this disk controller. In this case, when the access demand from a host computer was reference, data were transmitted to the host computer from the cache memory of the disk controller of a demand place, or the drive, or when the access demand from a host computer was updating, data were transmitted to the cache memory of the disk controller of a demand place. Therefore, in the case of this control system, coherence can be maintained, without carrying out complicated coherence control, since only the data of a drive connected to this disk controller will be stored in the cache of each disk controller.
[0057] A field is divided into the storing field of the data to the drive which connected the cache memory to the disk controller which received access, and the storing field of the data to the drive linked to other disk controllers in a subsystem, and it was made to manage. Consequently, a low cost cache memory with easy management can be offered further more well.
[0058] Since it doubled or multiplexed and the data to the drive linked to the disk controller which received access stored data on the cache memory, and they stored them on the other hand without multiplexing on the day memory of the others in a subsystem, cost can be reduced compared with the case where can realize higher reliability and all cache memories are doubled.
[0059] The disk controller ****** cache memory consisted of volatilization cache memories which store the data to the drive linked to the non-volatilized KYASSHI memory which stores the data to the drive linked to the disk controller which received access, and other disk controllers in a subsystem. Consequently, compared with the case, un-volatilizing-izing [all cache memories], the capacity of a non-volatilized cache memory with, more high cost can be reduced, and a low cost can be realized.
[0060] Since the data of a drive linked to this obstacle generating disk controller currently held into the cache of a normal disk controller were cancelled when an obstacle occurred in a certain disk controller in a subsystem, an obstacle does not spread at the time of an obstacle, either.
[0061] Also in a host computer, a part of connectable channel, and the subsystem that consists of a disk controller without the connecting means between disk controllers of exclusive use since it was made to be the switch which connects these channels, cache access of the means of communications between disk controllers is attained among two or more disk controllers.
[0062] As control information for controlling the cache memory which it had in the disk controller A channel, a disk controller, and the access log table holding the access frequency for every logic disk are prepared. It judges whether the logic disk of the channel with the highest access frequency among the channels which receive access to a certain logic disk, and this

access place is connected to the same disk controller. When not the same, the channel with this highest access frequency rearranged this logic disk on the drive of the connected disk controller. Moreover, when the same, the host computer which uses other channels which access this logic disk used the channel of the disk controller which connects this logic disk. Consequently, optimization of the data arrangement in a subsystem can be attained and it becomes possible to stop the operating frequency of the path between disk controllers low, and since the band required of the path between disk controllers can be stopped low,-izing can be carried out [low cost].

[Translation done.]

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]
[Drawing 1] It is an example of the block diagram showing the outline of the disk controller concerning this invention.
[Drawing 2] It is an example of the block diagram showing the cache control information of the disk controller concerning this invention.
[Drawing 3] It is the flow chart showing an example of the whole operation concerning this invention.
[Drawing 4] It is the flow chart showing an example of the updating access demand processing concerning this invention.
[Drawing 5] It is the flow chart showing an example of the coherence processing concerning this invention.
[Drawing 6] It is the flow chart showing an example of the coherence processing concerning this invention.
[Drawing 7] It is the flow chart showing an example of the reference access demand processing concerning this invention.
[Drawing 8] It is the flow chart showing an example of the reference access demand processing concerning this invention.
[Drawing 9] It is the flow chart showing an example of the cache management method concerning this invention.
[Drawing 10] It is the flow chart showing an example of the reference access demand processing concerning this invention.
[Drawing 11] It is an example of the block diagram showing the cache of the disk controller concerning this invention.
[Drawing 12] It is an example of the block diagram showing the cache of the disk controller concerning this invention.
[Drawing 13] It is the flow chart showing an example of the cache management method concerning this invention.
[Drawing 14] They are other examples of the block diagram showing the outline of the disk controller concerning this invention.
[Drawing 15] They are other examples of the block diagram showing the outline of the disk controller concerning this invention.
[Drawing 16] It is the flow chart showing an example of the data configuration method of the disk controller concerning this invention.
[Drawing 17] It is the block diagram showing the outline of the conventional disk controller concerning this invention.
[Drawing 18] It is the block diagram showing the outline of the conventional disk controller concerning this invention.
[Description of Notations]
0 — ... — a host computer and 1 — ... — a disk subsystem and 2 — ... — a channel and 3 — ... — control information and 4 — ... — storing data and 5 — ... — reference data and 6 — ... — updating data and 7 — ... — a logic disk and 10 — ... — a disk controller and 11 — ... — the

channel-control section and 12 — ... — a control memory and 13 — ... — the cache memory section and 14 — ... — a disk control section and 15 — ... — a joint

---

[Translation done.]

(19)日本国特許庁（ＪＰ）　(12)公開特許公報(A)　(11)特許出願公開番号
特開2002－123479
(P2002－123479A)
(43)公開日　平成14年４月26日(2002.4.26)

(51)Int.Cl.⁷	識別記号	FI		テーマコード(参考)
G06F 13/10	340	G06F 13/10	340B	5B005
3/06	301	3/06	301S	5B014
	304		304B	5B065
12/08	511	12/08	511Z	5B083
	531		531B	

審査請求　未請求　請求項の数14　ＯＬ　(全 21 頁)　最終頁に続く

(21)出願番号　特願2000－316257(P2000－316257)
(22)出願日　平成12年10月17日(2000.10.17)

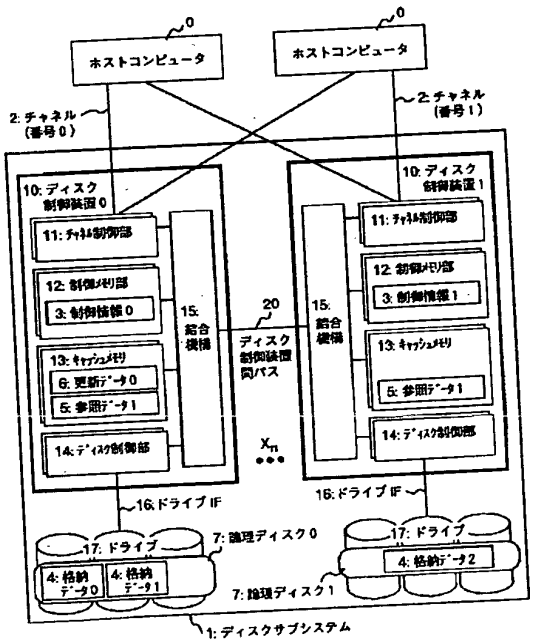
(71)出願人　000005108
株式会社日立製作所
東京都千代田区神田駿河台四丁目６番地
(72)発明者　金井　宏樹
東京都国分寺市東恋ヶ窪一丁目280番地
株式会社日立製作所中央研究所内
(72)発明者　藤本　和久
東京都国分寺市東恋ヶ窪一丁目280番地
株式会社日立製作所中央研究所内
(74)代理人　100068504
弁理士　小川　勝男　(外２名)

最終頁に続く

(54)【発明の名称】　ディスク制御装置およびそのキャッシュ制御方法

(57)【要約】
【課題】キャッシュを備えた複数のディスク制御装置において、装置間のキャッシュデーター致制御を行う。特定のディスク制御装置に障害が発生しても、他のディスク制御装置への障害伝播を防止する。
【解決手段】ディスク制御装置間の通信手段によりデータの一致制御を行う。ホストからの更新アクセスを受けた場合は、少なくともデータ格納ドライブを制御するディスク制御装置のキャッシュメモリはデータ更新を行う。望ましくは、キャッシュ領域を、当該ディスク制御装置が制御するドライブ用の領域と他のディスク制御装置が制御するドライブ用の領域とに分割して使用する。

図　1



【特許請求の範囲】
【請求項１】複数台のディスク制御装置と、ディスク制御装置間の通信手段と、ディスクドライブと、ディスク制御装置とディスクドライブ間に各々ディスクインターフェースとを備えたディスクサブシステムにおいて、前記ディスク制御装置は各々キャッシュメモリと該キャッシュメモリの制御情報を格納する制御メモリとを備え、ホストコンピュータからのアクセス要求を受領した一つのディスク制御装置に備えられたキャッシュメモリは、該キャッシュメモリを備えたディスク制御装置に前記ディスクインターフェースを介して接続されたディスクドライブに対するデータと、これに加えて、他の少なくとも一つのディスク制御装置に前記ディスクインターフェースを介して接続したディスクドライブに対するデータをも前記通信手段を介してアクセスし保持し得るようになしたことを特徴とするディスク制御装置。
【請求項２】制御メモリ部に格納する制御情報として、ディスク制御装置番号とディスクドライブアドレスから一意的に決定されるアクセス単位毎に、アクセス先のデータをキャッシメモリ上に保持しているディスク制御装置を特定するキャッシュディレクトリと、前記アクセス単位毎に該アクセス先のデータを格納するキャッシュアドレスとを保持することを特徴とする請求項１記載のディスク制御装置。
【請求項３】複数台のディスク制御装置と、ディスク制御装置間の通信手段と、ディスクドライブと、ディスク制御装置とディスクドライブ間に各々ディスクインターフェースとを備えたディスクサブシステムにおいて、前記ディスク制御装置は各々キャッシュメモリを備え、ホストコンピュータからのアクセス要求を受領したディスク制御装置は、アクセスデータの排他処理の後、アクセス要求を処理し、ホストへの完了報告を行い、その後に、ホストコンピュータからのアクセスが更新アクセス要求であり、かつ、アクセス受領ディスク制御装置以外のディスク制御装置が該アクセスデータをそのキャッシュメモリに保持している場合は、コヒーレンス制御を行った後に、該データの排他を解除することを特徴とするディスク制御装置のキャッシュメモリ制御方法。
【請求項４】ホストコンピュータから更新アクセス要求を受領したディスク制御装置は、該更新アクセス先が該ディスク制御装置以外の他のディスク制御装置にディスクインターフェースを介して接続したドライブに対する更新要求である場合は、ディスク制御装置間の通信手段を介してホストから受領した更新データを該ディスクドライブをディスクインターフェースを介して接続したディスク制御装置のキャッシュメモリに格納することを特徴とする請求項３記載のディスク制御装置のキャッシュメモリ制御方法。
【請求項５】ホストコンピュータから参照アクセス要求を受領したディスク制御装置は、アクセス先のディスクドライブをディスクインターフェースを介して接続するディスク制御装置の請求項２記載のキャッシュディレクトリを参照してアクセスデータがアクセス要求を受領したディスク制御装置内のキャッシュメモリに保持されているか判定し、該データが保持されている場合は、直ちに該キャッシュメモリを参照して該データをホストコンピュータに転送し、該キャッシュに保持されていない場合は、前記キャッシュディレクトリを参照してアクセスデータが該アクセス先のディスクドライブをディスクインターフェースを介して接続するディスク制御装置のキャッシュメモリに保持されているか判定し、該データが保持されている場合は、直ちに該キャッシュメモリを参照して該データをアクセス要求を受領したディスク制御装置内のキャッシュメモリとホストコンピュータに転送し、一方、前記キャッシュに保持されていない場合は、該アクセス先のディスクドライブから、該データを、該アクセス先のディスクドライブをディスクインターフェースを介して接続するディスク制御装置のキャッシュメモリと該アクセス要求を受領したディスク制御装置内のキャッシュメモリとホストコンピュータに転送することを特徴とする請求項４記載のディスク制御装置のキャッシュメモリ制御方法。
【請求項６】キャッシュメモリに保持した更新データをディスク制御装置にディスクインターフェースを介して接続するドライブに格納し、さらに、ディスクサブシステム内で該更新データを保持している別のディスク制御装置のキャッシュメモリの該更新データを無効化することを特徴とする請求項３乃至５のいずれかに記載のディスク制御装置のキャッシュメモリ制御方法。
【請求項７】複数台のディスク制御装置と、ディスク制御装置間の通信手段と、ディスクドライブと、ディスク制御装置とディスクドライブ間に各々ディスクインターフェースとを備えたディスクサブシステムにおいて、前記ディスク制御装置は各々キャッシュメモリを備え、該キャッシュメモリは、該ディスク制御装置にディスクインターフェースを介して接続したディスクドライブのデータのみを保持し、ホストコンピュータからのアクセス要求が参照の場合は、要求先のディスクドライブを接続したディスク制御装置のキャッシュメモリ、または、ディスクドライブから該データをホストコンピュータに転送し、あるいは、ホストコンピュータからのアクセス要求が更新の場合は、要求先のディスクドライブをディスクインターフェースを介して接続したディスク制御装置のキャッシュメモリに該データを転送することを特徴とするディスク制御装置のキャッシュメモリ制御方法。
【請求項８】キャッシュメモリの領域を、アクセスを受領したディスク制御装置にディスクインターフェースを介して接続したディスクドライブに対するデータの格納領域と、サブシステム内の他のディスク制御装置にディスクインターフェースを介して接続したディスクドライ

ブに対するデータの格納領域とに分割して管理することを特徴とする請求項３乃至７のいずれかに記載のディスク制御装置のキャッシュメモリ制御方法。

【請求項９】ホストコンピュータからのアクセスを受領したディスク制御装置にディスクインターフェースを介して接続したドライブに対するデータは、キャッシュメモリ上でデータを二重化、または、多重化して格納し、一方、サブシステム内の他のディスク制御装置にディスクインターフェースを介して接続したドライブに対するデータは、キャッシュメモリ上で多重化しないで格納することを特徴とする請求項３乃至８のいずれかに記載のディスク制御装置のキャッシュメモリ制御方法。

【請求項１０】複数台のディスク制御装置と、ディスク制御装置間の通信手段と、ディスクドライブと、ディスク制御装置とディスクドライブ間に各々ディスクインターフェースとを備えたディスクサブシステムにおいて、前記ディスク制御装置は各々キャッシュメモリを備え、該キャッシュメモリはサブシステム内の他のディスク制御装置にディスクインターフェースを介して接続したディスクドライブのデータをも保持可能とするキャッシュメモリ制御方法において、サブシステム内のあるディスク制御装置に障害が発生した場合は、正常なディスク制御装置のキャッシュメモリに保持している、該障害発生ディスク制御装置にディスクインターフェースを介して接続したディスクドライブのデータを無効化することを特徴とするディスク制御装置の制御方法。

【請求項１１】ディスク制御装置が備えるキャッシュメモリは、アクセスを受領したディスク制御装置にディスクインターフェースを介して接続したドライブに対するデータを格納する不揮発キャッシュメモリと、サブシステム内の他のディスク制御装置にディスクインターフェースを介して接続したドライブに対するデータを格納する揮発キャッシュメモリから構成することを特徴とする請求項１乃至２に記載のディスク制御装置。

【請求項１２】ディスク制御装置間の通信手段が、該ディスク制御装置内の相互結合網を拡張した結合網であることを特徴とする請求項１、２、１１のいずれかに記載のディスク制御装置。

【請求項１３】複数台のディスク制御装置と、ディスク制御装置間の通信手段と、ディスクドライブと、ディスク制御装置とディスクドライブ間に各々ディスクインターフェースとを備えたディスクサブシステムにおいて、前記ディスクドライブは、論理ディスクをその内部に有し、前記ディスク制御装置は、該ディスク制御装置内に備えたキャッシュメモリの制御情報として、チャネルとディスク制御装置と論理ディスクそれぞれに対するるアクセス頻度情報を保持することを特徴とする請求項１、２、１１、乃至１２のいずれかに記載のディスク制御装置。

【請求項１４】論理ディスクへのアクセスを受領するチャネルのうち、アクセス頻度が最も高いチャネルと該アクセス先の論理ディスクが同一のディスク制御装置にディスクインターフェースを介して接続されているかを判定し、同一でない場合は、該論理ディスクを該アクセス頻度が最も高いチャネルが接続されたディスク制御装置にディスクインターフェースを介して接続されているドライブ上に再配置し、同一である場合は、該論理ディスクにアクセスする他のチャネルを使用するホストコンピュータは、該論理ディスクを有するディスクドライブとディスクインターフェースを介して接続しているディスク制御装置のチャネルを使用することを特徴とする請求項１３に記載のディスク制御装置の制御方法。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】本発明は、多数台の磁気ディスク装置とこれらを制御するディスク制御装置から構成するディスクサブシステムに係わり、特に、ディスクサブシステムを複数台のディスク制御装置で構成するディスク制御装置、および、ディスク制御装置内キャッシュメモリの制御方法に関する。

【０００２】

【従来の技術】多数台の磁気ディスク装置（以下ディスクドライブあるいは単にドライブと呼ぶ）に対するデータの格納および読み出しを行うディスク制御装置（以下ＤＫＣと呼ぶ）がある。ドライブとＤＫＣは、あわせてディスクサブシステムと総称される。このような、従来のディスクサブシステムの構成を図１７に示す。

【０００３】本従来例では、２台のホストコンピュータ０がそれぞれ2台のディスクシステム１にチャネル２を介して接続されている。論理ディスク７は、ホストコンピュータ０が認識する記憶領域である。ホストコンピュータ０は、チャネル２を介して論理ディスク７の特定のアドレスに対してデータの参照、更新要求を指示する。チャネル２としては、ファイバチャネル、ＳＣＳＩなどがある。

【０００４】ディスクサブシステム１は、大きくは、ＤＫＣ１０と複数台のドライブ１７から構成される。ＤＫＣ１０と複数台のドライブ１７は、ドライブインターフェース（以下ドライブＩＦと呼ぶ）１６で接続される。ドライブＩＦ１６には、ファイバチャネル、ＳＣＳＩなどが用いられる。

【０００５】ＤＫＣ１０は、大きくは、チャネルの制御を行うチャネル制御部１１、ドライブの制御を行うディスク制御部１４、ＤＫＣの制御情報３を格納する制御メモリ部１２、参照データ５、更新データ６を保持するキャッシュメモリ部１３、さらに、各構成部品を相互に接続する結合機構１５から構成される。結合機構１５は、バス、相互結合網などが用いられる。

【０００６】ＤＫＣ１０は、ホストコンピュータ０の指示に従い、データの参照、更新処理を行う。このよう

な、従来のディスクサブシステムは、例えば、特開平２０００－９９２８１に開示されている。

【０００７】インターネットの爆発的な普及に代表されるように、近年のコンピュータ環境では、ユーザの使用する記憶容量は急激に増大している。この結果、日々増大するデータの管理コストも増大の一途をたどり、この管理コストの削減が重要課題となっている。また、従来サーバ毎に接続され、この結果分散配置されていたディスクサブシステムの集中化を図るべくストレージエリアネットワーク（以下ＳＡＮと呼ぶ）が注目されている。図１８は、ＳＡＮ環境におけるディスクサブシステムの従来例である。複数台のサブシステムがＳＡＮ－ＳＷ１１を介してホストコンピュータ０に接続される。１つのディスクサブシステムは１台のディスク制御装置のみから構成されている。

【０００８】以上説明したようにディスクサブシステムを取り巻く環境の変化により、ディスクサブシステムには、より一層の記憶容量増大と接続チャネル数の増大が要求されている。

【０００９】このような背景から、従来一台のＤＫＣから構成していたディスクサブシステムを、複数台のＤＫＣで構成することが考えられる。これにより、ディスクサブシステムとして、より大きな記憶容量と接続チャネル数を提供可能となる。

【００１０】複数台のＤＫＣを構成する一般的な方法として、ＤＫＣをクラスタ構成にすることが考えられる。しかし、この場合ＤＫＣ間でのデータの共有が困難となるという課題がある。この課題を解決するために、ＤＫＣ間の接続手段を用いて、ＤＫＣ間で相互にデータのアクセスを可能とすることでデータの共有を実現できる。

【００１１】

【発明が解決しようとする課題】しかし、相互にデータアクセス可能な複数台のＤＫＣからなるディスクサブシステムでは、各ＤＫＣ間にあるキャッシュメモリ間のデーター致問題が重要である。これは、一般的には、コヒーレンス制御と呼ばれる。

【００１２】特に、ディスクサブシステムの場合、ドライブのデータがユーザデータの最終記憶部であるため、このデータを保証することが重要であり、例えば、ディスクサブシステム内の一台のＤＫＣに障害が発生しダウンした場合でも、他のＤＫＣに障害が伝播してはならない。しかしながら、障害が発生したＤＫＣのキャッシュメモリに他の正常動作可能なＤＫＣに接続したドライブのデータが更新された状態で残されている場合には、このデータに対するアクセスが不能となるため、このデータを格納するドライブとこのドライブを管理するＤＫＣが正常にも係わらず、データロストが生じてしまう。つまり、サブシステム内の1台のＤＫＣの障害が同じサブシステム内の他のＤＫＣに伝播することとなり問題となる。

【００１３】

【課題を解決するための手段】各ディスク制御装置は、ホストからのアクセス要求を受領したディスク制御装置に接続したドライブに対するデータに加えて、該ディスク制御装置間の通信手段を介してホストからのアクセス要求を受領したディスク制御装置以外の他のディスク制御装置に接続したドライブに対するデータを保持するキャッシュメモリと該キャッシュメモリの制御情報を格納する制御メモリとを備えるようにした。

【００１４】さらに、該ディスク制御装置は、ディスク制御装置内に備えたキャッシュを制御するための制御情報として、ディスク制御装置とドライブのアドレスから一意に決定可能なアクセス単位毎に、アクセス先のデータを参照しキャッシュ上に保持しているディスク制御装置を特定できるキャッシュディレクトリと、該アクセス先のデータを格納するキャッシュアドレスとを保持するキャッシュ管理テーブルを設けるようにした。

【００１５】ホストコンピュータからのアクセス要求を受領したディスク制御装置は、処理の始めにアクセスデータの排他処理を行い、その後、アクセス要求を処理しホストへの完了報告を行った後に、ホストコンピュータからのアクセスが更新アクセス要求であり、かつ、アクセス受領ディスク制御装置以外のディスク制御装置が該アクセスデータをそのキャッシュメモリに保持している場合は、コヒーレンス制御を行った後に、該データの排他を解除するようにした。

【００１６】ホストコンピュータから更新アクセス要求を受領したディスク制御装置は、該更新アクセス先が該ディスク制御装置以外の他のディスク制御装置に接続したドライブに対する更新要求である場合は、ディスク制御装置間の通信手段を介してホストコンピュータから受領した更新データを該ドライブを接続したディスク制御装置のキャッシュメモリに格納するようにした。

【００１７】コヒーレンス制御方法は、他のディスク制御装置のキャッシュに保持しているデータを無効化、あるいは、他のディスク制御装置のキャッシュメモリに保持しているデータを更新するようにした。

【００１８】ホストコンピュータから参照アクセス要求を受領したディスク制御装置は、始めに、アクセス先のドライブを接続するディスク制御装置のキャッシュ管理テーブルのディレクトリを参照してアクセスデータがアクセス要求を受領したディスク制御装置内のキャッシュに保持されているか判定し、該データが保持されている場合は、直ちに該キャッシュを参照して該データをホストコンピュータに転送し、一方、該アクセスデータがアクセス要求を受領したディスク制御装置内のキャッシュに保持されていない場合は、アクセス先のドライブを接続するディスク制御装置のキャッシュ管理テーブルのディレクトリを参照してアクセスデータが該アクセス先のドライブを接続するディスク制御装置のキャッシュメモ

リに保持されているか判定し、該データが保持されている場合は、直ちに該キャッシュメモリを参照して該データをアクセス要求を受領したディスク制御装置内のキャッシュとホストコンピュータに転送し、一方、該アクセス先のドライブを接続するディスク制御装置のキャッシュメモリに保持されていない場合は、ドライブから、該データを、該アクセス先のドライブを接続するディスク制御装置のキャッシュメモリと該アクセス要求を受領したディスク制御装置内のキャッシュメモリとホストコンピュータに転送するようにした。

【００１９】キャッシュ領域の解放する場合は、該キャッシュに保持した更新データを該ディスク制御装置に接続するドライブに格納し、さらに、ディスクサブシステム内で該データを保持している別のディスク制御装置のキャッシュメモリの該データを無効化するようにした。

【００２０】各ディスク制御装置に備えたキャッシュメモリは、該ディスク制御装置に接続したドライブのデータのみを保持することにより、ホストコンピュータからのアクセス要求が参照の場合は、要求先のディスク制御装置のキャッシュ、または、ドライブからデータをホストに転送し、あるいは、ホストコンピュータからのアクセス要求が更新の場合は、要求先のディスク制御装置のキャッシュにデータを転送するようにした。

【００２１】キャッシュメモリの格納領域は、アクセスを受領したディスク制御装置に接続したドライブに対するデータの格納領域と、サブシステム内の他のディスク制御装置に接続したドライブに対するデータの格納領域とに領域を分割して管理するようにした。

【００２２】アクセスを受領したディスク制御装置に接続したドライブに対するデータは、キャッシュメモリ上でデータを二重化、または、多重化して格納し、一方、サブシステム内の他のディスク制御装置に接続したドライブに対するデータは、キャッシュ上で多重化しないで格納するようにした。

【００２３】ディスク制御装置に備えるキャッシュは、アクセスを受領したディスク制御装置に接続したドライブに対するデータを格納する不揮発キャッシュと、サブシステム内の他のディスク制御装置に接続したドライブに対するデータを格納する揮発キャッシュから構成するようにした。

【００２４】サブシステム内のあるディスク制御装置に障害が発生した場合は、正常なディスク制御装置のキャッシュメモリに保持している、該障害発生ディスク制御装置に接続したドライブのデータは無効化するようにした。

【００２５】ディスク制御装置間の通信手段は、ホストコンピュータと接続が可能なチャネルの一部と、該チャネル同士を接続するスイッチであるようにした。

【００２６】ディスク制御装置内に備えたキャッシュを制御するための制御情報として、チャネルとディスク制御装置と論理ディスク毎のアクセス頻度情報を保持するアクセスログテーブルを設け、ある論理ディスクへのアクセスを受領するチャネルのうち、アクセス頻度が最も高いチャネルと該アクセス先の論理ディスクが同一のディスク制御装置に接続されているかを判定し、同一でない場合は、該論理ディスクを該アクセス頻度が最も高いチャネルが接続されたディスク制御装置のドライブ上に再配置するようにした。また、同一である場合は、該論理ディスクにアクセスする他のチャネルを使用するホストコンピュータは、該論理ディスクを接続するディスク制御装置のチャネルを使用するようにした。

【００２７】

【発明の実施の形態】以下、図面を用いて、発明の詳細を説明する。始めに、図１、および、図２を用いて、本発明に係わるディスク御装置について説明する。図1は、本発明に係わるディスク制御装置の概要を示すブロック図の一例である。ディスクサブシステム１は、複数のチャネル２を介して、ホストコンピュータ０に接続される。本実施例では、ディスクサブシステム１を複数台のＤＫＣ１０により構成し、各ＤＫＣ１０は、各ＤＫＣ１０に専用のディスク制御装置間パス２０を介して、他のＤＫＣに接続したドライブ１７に格納した格納データ４を参照、更新できるところに特徴がある。以下、詳細に説明する。

【００２８】図１に示したディスクサブシステム１は、大きくは、複数台のＤＫＣ１０と、ドライブ１７から構成する。ＤＫＣ１０の詳細は２台のみ詳細に示しているが、Ｘｎ台の各ＤＫＣ１０の各々の構成は同一である。ＤＫＣ１０は、大きくは、チャネルの制御を行うチャネル制御部１１、ドライブの制御を行うディスク制御部１４、ＤＫＣの制御情報３を格納する制御メモリ部１２、参照データ５、更新データ６を保持するキャッシュメモリ部１３、さらに、各構成部品を相互に接続する結合機構１５から構成する。図示はしていないが、チャネル制御部１１やディスク制御部１４は、制御用のプロセサを備え、プロセサ上で処理プログラムが動作する。

【００２９】図２は、図１における制御メモリ１２に格納する制御情報の一例を示したブロック図である。本実施例では、制御情報３は、大きくは、キャッシュ管理テーブル３１と装置構成管理テーブルから構成する。

【００３０】まず、キャッシュ管理テーブル３１について詳述する。キャッシュ管理テーブル３１は、ディレクトリ情報とキャッシュアドレス情報を保持している。本実施例では、説明上、ディレクトリ情報とキャッシュアドレス情報を個別のテーブルとして記述するが、これらは同一のテーブルとしても良い。ディレクトリ情報は、ホストのアクセス先ドライブアドレスと、そのアドレスのデータを保持しているキャッシュの関係を示す。具体的には、ホストのアクセス先アドレスとして、アクセス先のディスク制御装置番号とドライブアドレスを持ち、

さらに、サブシステム内の各キャッシュがそのアドレスに対するデータを保持しているかどうかを各キャッシュ毎に示すディレクトリである。本実施例では、キャッシュディレクトリは、ＤＫＣ毎に用意してあり、キャッシュディレクトリが１の場合は、そのＤＫＣのキャッシュがデータを保持していることを、また、キャッシュディレクトリが０の場合は、そのＤＫＣのキャッシュがデータを保持していないことを示している。従って、ディスク制御装置番号０、ドライブアドレス０のデータは、ＤＫＣ０のキャッシュに格納されていることを示している。また、ディスク制御装置番号０、ドライブアドレス１のデータは、ＤＫＣ０のキャッシュとＤＫＣ１のキャッシュ両方に格納されていることを示している。

【００３１】次にキャッシュアドレス情報であるが、これは、ホストのアクセス先ドライブアドレスと、そのアドレスのデータを保持しているキャッシュアドレスの関係を示す。ディレクトリ情報同様にホストコンピュータのアクセス先アドレスとして、アクセス先のディスク制御装置番号とドライブアドレスを持ち、そのアドレスに対するデータを保持しているキャッシュアドレスを示している。キャッシュアドレス情報は、ＤＫＣ内のキャッシュに対するアドレスのみ保持しても良い。

【００３２】次に装置構成管理テーブルについて詳述する。装置構成管理テーブルは、ホストコンピュータが識別可能なチャネル番号および論理ディスクと、ＤＫＣ内の実際のデータ格納先ドライブとの関係を示す。実施例では、チャネル番号１の論理ディスク０は、ディスク制御装置番号０のドライブ番号０に割当てられていることを示す。また、チャネル番号１の論理ディスク１は、ディスク制御装置番号１のドライブ番号１に割当てられていることを示す。

【００３３】以上説明した、制御情報３を用いると、装置構成管理テーブル３２を参照することでホストのアクセス要求先のディスク制御装置番号とドライブ番号を識別可能であり、さらに、キャッシュ管理テーブル３１のディレクトリ情報を参照することでアクセス先のデータを保持しているキャッシュメモリ番号を識別可能であり、さらに、該当するキャッシュメモリのキャッシュアドレス情報を参照すことでアクセス先のデータを保持しているキャッシュアドレスを理解することが可能である。

【００３４】本実施例では、制御メモリ部１２に、装置構成管理テーブル３２を格納しているが、該装置構成管理テーブル３２の格納先は、チャネル制御部１１やディスク制御部１４に備えた制御プロセサのローカルメモリ上に格納しても良い。

【００３５】次に図３から図１０に示す流れ図を用いてホストコンピュータからのアクセス要求を受領した場合の、ＤＫＣの動作およびキャッシュメモリ制御方法について説明する。

【００３６】始めに、図３を用いて処理の概要を示す。図３は、ＤＫＣの処理全体の流れを示した流れ図である。本流れ図で示した処理は、ＤＫＣ内のプロセサ上の処理プログラムとして実現できる。ホストコンピュータからのアクセス要求を受領したＤＫＣは、始めに受信コマンドの処理とアクセスデータの排他処理を行う（ステップ１）。接続チャネルのプロトコルに従い、コマンド解析することで、アクセス要求がリードコマンド、すなわち、 参照要求であるか、あるいは、ライトコマンド、すなわち、更新要求であるかを識別できる。さらに、アクセス先データが他のアクセス処理によって更新、参照されないように排他処理を行う。排他処理は、ロックなどによる一般的な方法で行えば良い。次に、ＤＫＣは、コマンドに応じて参照、あるいは、更新処理を行う（ステップ２）。次に、処理終了後、ホストへの完了報告を行い、（ステップ３）。受領コマンドの判定を行う（ステップ４）。ライトコマンドでない場合は後述のステップ７に進む。ライトコマンドの場合は、次に、ホストがアクセスした当該アドレスのデータをサブシステム内の他のＤＫＣがキャッシュメモリに保持可能かを判定する（ステップ５）。他のＤＫＣがキャッシュメモリに保持できない場合は後述のステップ７に進み、処理が終了となる。他のＤＫＣがキャッシュに保持可能な場合は、更新データのコヒーレンス処理を行う（ステップ６）。最後に、アクセス先のデータの排他を解除する（ステップ７）。本実施例では、ライトコマンド処理時に、他のＤＫＣがキャッシュ上に旧データを保持している場合は、データのコヒーレンス処理を行うところに特徴がある。コマンド処理、あるいは、コヒーレンス処理の詳細はそれぞれ後述する。

【００３７】図４は、ライトコマンド受領時の処理の一例を示す流れ図である。コマンドを受領すると、受信コマンドから、コマンドとアクセス先アドレスを解析し，ライトアクセスであることを認識する（ステップ１）。アクセス先アドレスは、装置構成管理テーブルを参照することで、アクセス要求先のディスク制御装置番号とドライブ番号を識別できる。次に、ステップ１で識別した当該ＤＫＣのキャッシュメモリに対してキャッシュヒットミス判定を行う（ステップ２）。キャッシュ管理テーブルのディレクトリ情報を参照することで、アクセス先データがキャッシュに保持されているかを識別可能である。キャッシュに保持していないキャッシュミスの場合は、当該ＤＫＣのディスク制御部対して当該データのドライブからキャッシュメモリへの転送依頼を行う（ステップ６）。通常この処理はステージング処理と呼ばれる。この場合転送終了までライト処理を中断し（ステップ７）、ステージング終了後、再びライト処理を継続することになる。また、転送先のキャッシュアドレスは、キャッシュの空きリストなど一般的な方法で管理、取得すればよいが、転送先アドレスをキャッシュ管理テーブ

ルを更新することで登録する必要がある。ステップ３でヒット判定の場合、または、ステップ７でステージング処理が終了した場合は、当該ＤＫＣのキャッシュメモリに対して当該データの更新を行う（ステップ４）。更新終了後、ホストコンピュータに対してライト処理の完了報告を行う（ステップ５）。本実施例では、キャッシュ管理テーブルのキャッシュディレクトリとキャッシュアドレスを参照することでサブシステム内の全てのＤＫＣのキャッシュメモリをアクセスできるところに特徴がある。

【００３８】図５は、ライトコマンド受領時のライト処理に続いて行うキャッシュメモリのコヒーレンス制御の一例を示す流れ図である。本実施例では、他のキャッシュメモリのデータを無効化するところに特徴がある。キャッシュ管理テーブルのディレクトリ情報を参照することで更新要求アドレスの旧データを保持している他のＤＫＣのキャッシュメモリがあるかを判定する（ステップ１）。旧データを保持している他のＤＫＣのキャッシュメモリが無い場合は終了である。旧データを保持している他のＤＫＣのキャッシュメモリがある場合は、ディレクトリ情報を更新する。例えば、実施例では、保持状態を１で示しているので、ディレクトリ情報として０を書き込めば良い。さらに、当該キャッシュメモリの旧データ保持領域を解放する必要がある。当該キャッシュメモリの旧データ保持アドレスは、当該ＤＫＣのキャッシュアドレス情報を参照することで認識できる。該キャッシュアドレス情報から、該キャッシュアドレスを削除し、該キャッシュメモリ領域を前述のキャッシュメモリの空きリストば良い。以上により、データのコピーを保持している他のＤＫＣのキャッシュメモリを無効化する（ステップ２）。

【００３９】図６は、ライトコマンド受領時のライト処理に続いて行うキャッシュメモリのコヒーレンス制御の他の一例を示す流れ図である。本実施例では、他のキャッシュメモリのデータを更新するところに特徴がある。キャッシュ管理テーブルのディレクトリ情報を参照することで更新要求アドレスの旧データを保持している他のＤＫＣのキャッシュメモリがあるかを判定する（ステップ１）。旧データを保持している他のＤＫＣのキャッシュメモリが無い場合は終了である。旧データを保持している他のＤＫＣのキャッシュメモリがある場合は、データを更新する。当該キャッシュの旧データ保持アドレスは、当該ＤＫＣのキャッシュアドレス情報を参照することで認識できる。該キャッシュアドレスに対して更新データを書き込めば良い。以上により、当該データのコピーを保持している他のＤＫＣのキャッシュを更新する（ステップ２）。

【００４０】図７、図８は、リードコマンド受領時の処理の一例を示す流れ図である。図では受領コマンドを受信コマンドとして表わしてある。本実施例では、ＤＫＣ間で他ＤＫＣの管理するデータのキャッシュメモリへの保持が可能な場合について示す。コマンドを受領すると、受信コマンドから、コマンドとアクセス先アドレスを解析し、リードアクセスであることを認識する（ステップ１）。アクセス先アドレスは、装置構成管理テーブルを参照することで、アクセス要求先のディスク制御装置番号とドライブ番号を識別できる。次に、ステップ１で識別した当該ＤＫＣのキャッシュメモリに対してキャッシュヒットミス判定を行う（ステップ２）。キャッシュ管理テーブルのディレクトリ情報を参照することで、アクセス先データがキャッシュメモリに保持されているかを識別可能である。コマンド受領ＤＫＣのキャッシュメモリに保持しているか判定し（ステップ３）、コマンド受領ＤＫＣのキャッシュメモリに保持している場合は、直ちに、該コマンド受領ＤＫＣのキャッシュメモリに対して当該データを参照する（ステップ４）。さらに、当該データをチャネルに転送する（ステップ５）。一方、ステップ３で、キャッシュミスの場合は、アクセス先のドライブを接続するＤＫＣのキャッシュメモリに保持しているかを判定する（ステップ６）。アクセス先のドライブを接続するＤＫＣのキャッシュメモリに保持している場合は、該アクセス先のドライブを接続するＤＫＣのキャッシュメモリから、コマンド受領ＤＫＣのキャッシュメモリにデータを転送する。この際、該コマンド受領ＤＫＣのキャッシュメモリのデータ格納先アドレスは、前述のキャッシュメモリの空きリストなどから取得すれば良いが、該アドレスを該コマンド受領ＤＫＣのキャッシュ管理テーブルのキャッシュアドレス情報に書き込む必要がある。更に、アクセス先のドライブを接続するＤＫＣのキャッシュ管理テーブルのディレクトリ情報は、該コマンド受領ＤＫＣのキャッシュにデータをコピーしたことを示すように更新する（ステップ７）。転送終了待ち（ステップ８）の後、先述のステップ４に進む。一方、ステップ６でキャッシュミスとなった場合は、ドライブから、データを読み込む必要がある。通常この処理はステージング処理と呼ばれ（ステップ９）、本ステップは、図４のステップ６と同様である。転送終了待ち（ステップ１０）の後、先述のステップ４に進む。

【００４１】図９は、ＤＫＣ間でデータのキャッシュメモリへの保持が可能な場合の、キャッシュ領域の解放方法の一例を示す流れ図である。キャッシュに空き領域が無くなった場合は、所定のアルゴリズムにしたがって、キャッシュの領域を解放する必要がある。一般的なアルゴリズムとしては、ＬＲＵ法がある。この方法では、始めに所定のアルゴリズムにしたがって、解放する領域を決定する。この後、解放領域に現在格納しているデータが自ＤＫＣに接続するドライブのデータか否かを判定する（ステップ１）。この判定は、キャッシュ管理テーブルのキャッシュアドレス情報を参照することにより行う

ことができる。自ＤＫＣに接続するドライブのデータでない場合は、当該データ格納先ドライブを接続、管理するＤＫＣのキャッシュ管理テーブルのディレクトリ情報を更新し、自ＤＫＣのキャッシュメモリがデータを保持していないようにする（ステップ５）。一方、ステップ１で、自ＤＫＣに接続するドライブのデータであると判定した場合は、ディスク制御部に対して該当データのドライブへの書き込みを依頼する（ステップ２）。本処理は、通常デステージ処理と呼ばれる。書き込み終了を待った後（ステップ３）、キャッシュ管理テーブルのディレクトリ情報に従って、本データのコピーを保持している他のＤＫＣのキャッシュ対して当該データを無効化する（ステップ４）。

【００４２】図１０は、リードコマンド受領時の処理の一例を示す流れ図である。本実施例では、ＤＫＣ間で他ＤＫＣの管理するデータのキャッシュメモリへの保持が不可能な場合について示す。コマンドを受領すると、受信コマンドから、コマンドとアクセス先アドレスを解析し、リードアクセスであることを認識する（ステップ１）。アクセス先アドレスは、装置構成管理テーブルを参照することで、アクセス要求先のディスク制御装置番号とドライブ番号を識別できる。次に、ステップ１で識別した当該ＤＫＣのキャッシュメモリに対してキャッシュヒットミス判定を行う（ステップ２）。キャッシュ管理テーブルのディレクトリ情報を参照することで、アクセス先データがキャッシュに保持されているかを識別可能である。アクセス先のドライブを接続するＤＫＣのキャッシュメモリに保持しているか判定し（ステップ３）、アクセス先のドライブを接続するＤＫＣのキャッシュに保持している場合は、直ちに、該アクセス先のドライブを接続するＤＫＣのキャッシュに対して当該データを参照する（ステップ４）。さらに、当該データをチャネルに転送する（ステップ５）。一方、ステップ３で、キャッシュミスの場合は、ドライブから、データを読み込む必要がある。通常この処理はステージング処理と呼ばれ（ステップ６）、本ステップは、図４のステップ６と同様である。転送終了待ち（ステップ７）の後、先述のステップ４に進む。

【００４３】次に、図１１を用いて、キャッシュメモリ部１３の管理方法の望ましい他の一例について説明する。本実施例では、キャッシュメモリ部１３の領域を、他ＤＫＣ用データ格納領域と、自ＤＫＣ用データ格納領域とに分割したところに特徴がある。この結果、後述する、データの二重化乃至多重化、あるいは、キャッシュの一部不揮発化を、容易かつ低コストで実現可能となる。領域を分割するためには、キャッシュメモリの空き領域を管理するリストが各領域毎に必要である。

【００４４】本実施例では、自ＤＫＣ用データ格納領域のみ二重化している。ホストからの更新データは、該データの格納先ドライブの接続するＤＫＣのキャッシュメモリ、すなわち、自ＤＫＣ用データ格納領域に保持する。従って、該自ＤＫＣ用データ格納領域を二重化することで信頼性を向上できる。また、該自ＤＫＣ用データ格納領域のみを二重化することでキャッシュメモリ部１３全体を二重化する場合と比較して、低コストで信頼性を確保できる。

【００４５】次に、図１２を用いて、キャッシュメモリ部１３の管理方法の望ましい他の一例について説明する。本実施例では、キャッシュメモリ部１３を、他ＤＫＣ用データを格納する揮発キャッシュ領域１３１と、自ＤＫＣ用データ格納する不揮発キャッシュ領域１３２とから構成することに特徴がある。

【００４６】次に図１５、図16を用いて、ディスク制御装置の他の一例を示す。図１５は、図１における制御メモリ１２に格納する制御情報の他の一例を示したブロック図である。本実施例では、制御情報として、キャッシュ管理テーブルと装置構成管理テーブルの他にホストからのアクセス回数の統計を示すアクセスログテーブル３３を備えたところに特徴がある。アクセスログテーブル３３は、チャネル番号、論理ディスク番号、ディスク制御装置番号ごとにアクセス回数を示す。本実施例では、リード回数、ライト回数に分けて保持している。アクセス回数は、ホストからの受領コマンド解析時、あるいは、キャッシュヒットミス判定時に、各処理プログラムが、制御情報をアクセスする際に、あわせて、回数をインクリメント更新するようにすれば良い。サブシステム内の各ＤＫＣのアクセスログテーブル３３を解析することでホストのアクセス特性を認識できる。本実施例では、論理ディスク番号０番へのアクセスは、チャネル番号０、１、３からアクセスされており、チャネル１からのアクセスはディスク制御装置間パス２０を介するデータ転送が必要であることがわかる。

【００４７】次に、具体的なアクセス頻度の識別方法例を図１６を用いて説明する。図１６は、ＤＫＣ内のプロセサ上の処理プログラムが実行するアクセス特性認識方法の処理の流れ図である。本実施例では、ＤＫＣ内のプロセサ上の処理プログラムが実行することを想定するが、ＤＫＣ外の管理用プロセサ上の処理プログラムが実行しても良い。アクセス特性認識処理は、タイマにより定期的に実行、あるいは、ホストからの指示に従って実行するようにすれば良い。始めに、各ディスク制御装置のアクセスログテーブル３３から、特定のディスク制御装置の論理ディスクへのアクセスについて、チャネル番号毎のアクセス回数を抽出し、チャネル毎のアクセス回数を比較する（ステップ1）。次に、抽出、比較したアクセス回数のうち、アクセス回数が最大のチャネルが接続されたＤＫＣ番号が、解析対象の論理ディスクが接続されたＤＫＣ番号と同一かを判定する（ステップ２）。本判定により、論理ディスクへのアクセス頻度が高いチャネルが、論理ディスクと同一のＤＫＣに接続されてい

るかを判定する。ステップ２の判定で同一である場合は、アクセス回数が最大のチャネルと論理ディスクは同一ＤＫＣに接続されているので、論理ディスクの再配置は不要であり、該論理ディスクをアクセスしている他のチャネルのうちで、ディスク制御装置間パス２０を用いている必要のあるチャネルは、論理ディスクを接続しているＤＫＣのチャネルを使用するようにする（ステップ3）。指示は、ＤＫＣの構成管理用プロセサ、または、ホストコンピュータに対して行えば良い。一方、ステップ2の判定で、同一でない場合は、アクセス回数が最大のチャネルと論理ディスクは同一ＤＫＣに接続されていないので、該チャネルと該論理ディスクが同一のＤＫＣに接続されるようにすると良い。本実施例では、当該論理ディスクを、アクセス回数が最大のチャネルを接続するＤＫＣのドライブに再配置するように指示する（ステップ4）。ステップ１からステップ4をサブシステム内の全ての論理ディスクについて繰り返し行うことで、ディスク制御装置間パスの使用頻度を低くし、ホストからのアクセスをアクセスを受領したＤＫＣの論理ディスクへのアクセスにすることができるので、ディスク制御装置間パスの帯域が低くても性能を維持できるように成る。

【００４８】

【発明の効果】各ディスク制御装置は、ホストからのアクセス要求を受領したディスク制御装置に接続したドライブに対するデータに加えて、該ディスク制御装置間の通信手段を介してホストからのアクセス要求を受領したディスク制御装置以外の他のディスク制御装置に接続したドライブに対するデータを保持するキャッシュメモリと該キャッシュメモリの制御情報を格納する制御メモリとを備えるようにしたので、各ディスク制御装置内に備えたキャッシュ間で、データの共有が可能となり、性能向上できる。

【００４９】さらに、該ディスク制御装置は、ディスク制御装置内に備えたキャッシュメモリを制御するための制御情報として、ディスク制御装置とドライブのアドレスから一意に決定可能なアクセス単位毎に、アクセス先のデータを参照しキャッシュメモリ上に保持しているディスク制御装置を特定できるキャッシュディレクトリと、該アクセス先のデータを格納するキャッシュアドレスとを保持するキャッシュ管理テーブルを設けるようにしたので、キャッシュメモリのコヒーレンス制御が可能となる。

【００５０】ホストコンピュータからのアクセス要求を受領したディスク制御装置は、処理の始めにアクセスデータの排他処理を行い、その後、アクセス要求を処理しホストへの完了報告を行った後に、ホストコンピュータからのアクセスが更新アクセス要求であり、さらに、アクセス受領ディスク制御装置以外のディスク制御装置が該アクセスデータをキャッシュに保持している場合は、コヒーレンス制御を行った後に、該データの排他を解除するようにしたので、ホストコンピュータの応答時間を増大することなくコヒーレンス制御を実現できる。

【００５１】ホストコンピュータから更新アクセス要求を受領したディスク制御装置は、該更新アクセス先が該ディスク制御装置以外の他のディスク制御装置に接続したドライブに対する更新要求である場合は、ディスク制御装置間の通信手段を介してホストから受領した更新データを該ドライブを接続したディスク制御装置のキャッシュメモリに格納するようにしたので、ディスクサブシステム内のあるディスク制御装置に障害が発生した場合でも、他のディスク制御装置のデータはロストすることなく障害の伝播を防止できる。

【００５２】コヒーレンス制御方法は、他のディスク制御装置のキャッシュメモリに保持しているデータを無効化するようにしたので、ディスク制御装置間の接続手段の転送帯域が低い場合でも、コヒーレンス制御が可能となる。

【００５３】また、別のコヒーレンス制御方法としては、他のディスク制御装置のキャッシュメモリに保持しているデータを更新するようにしたので、よりキャッシュメモリのヒット率が向上し、性能が改善される。

【００５４】ホストコンピュータから参照アクセス要求を受領したディスク制御装置は、始めに、アクセス先のドライブを接続するディスク制御装置のキャッシュ管理テーブルのディレクトリを参照してアクセスデータがアクセス要求を受領したディスク制御装置内のキャッシュメモリに保持されているか判定する。該データが保持されている場合は、直ちに該キャッシュメモリを参照して該データをホストコンピュータに転送する。一方、該アクセスデータがアクセス要求を受領したディスク制御装置内のキャッシュメモリに保持されていない場合は、アクセス先のドライブを接続するディスク制御装置のキャッシュ管理テーブルのディレクトリを参照してアクセスデータが該アクセス先のドライブを接続するディスク制御装置のキャッシュメモリに保持されているか判定する。該データがそこに保持されている場合は、直ちに該キャッシュメモリを参照して該データをアクセス要求を受領したディスク制御装置内のキャッシュメモリとホストコンピュータに転送する。一方、該アクセス先のドライブを接続するディスク制御装置のキャッシュメモリに保持されていない場合は、ドライブから、該データを、該アクセス先のドライブを接続するディスク制御装置のキャッシュメモリと該アクセス要求を受領したディスク制御装置内のキャッシュメモリとホストコンピュータに転送するようにした。したがって、アクセス要求を受領したディスク制御装置以外のディスク制御装置に接続したドライブのデータであっても、参照が可能であり、さらに、該アクセスデータが、キャッシュメモリに保持されている場合は、ドライブにアクセスする場合に比べ短い応答時間でホストコンピュータにデータを転送するこ

とができる。

【００５５】キャッシュ領域を解放する場合は、該キャッシュメモリに保持した更新データを該ディスク制御装置に接続するドライブに格納し、さらに、ディスクサブシステム内で該データを保持している別のディスク制御装置のキャッシュの該データを無効化するようにしたので、キャッシュを効率よく使用できる。

【００５６】各ディスク制御装置に備えたキャッシュメモリは、該ディスク制御装置に接続したドライブのデータのみを保持することにした。その場合、ホストコンピュータからのアクセス要求が参照の時は、要求先のディスク制御装置のキャッシュメモリ、または、ドライブからデータをホストコンピュータに転送し、あるいは、ホストコンピュータからのアクセス要求が更新の時は、要求先のディスク制御装置のキャッシュメモリにデータを転送するようにした。したがって、本制御方式の場合は、各ディスク制御装置のキャッシュには、該ディスク制御装置に接続されたドライブのデータのみ格納することとなるため、複雑なコヒーレンス制御をすることなく，コヒーレンスを維持することができる。

【００５７】キャッシュメモリを、アクセスを受領したディスク制御装置に接続したドライブに対するデータの格納領域と、サブシステム内の他のディスク制御装置に接続したドライブに対するデータの格納領域とに領域を分割して管理するようにした。その結果、管理が容易な、さらに、より効率の良い、あるいは、低コストなキャッシュメモリを提供できる。

【００５８】アクセスを受領したディスク制御装置に接続したドライブに対するデータは、キャッシュメモリ上でデータを二重化、または、多重化して格納し、一方、サブシステム内の他のディメモリ上で多重化しないで格納するようにしたので、より高い信頼性を実現でき、かつ、全キャッシュメモリを二重化する場合に比べコストを低減できる。

【００５９】ディスク制御装置備えるキャッシュメモリは、アクセスを受領したディスク制御装置に接続したドライブに対するデータを格納する不揮発キャッシメモリと、サブシステム内の他のディスク制御装置に接続したドライブに対するデータを格納する揮発キャッシュメモリから構成するようにした。その結果、全キャッシュメモリを不揮発化する場合に比べ、よりコストの高い不揮発キャッシュメモリの容量を低減でき、低コストを実現できる。

【００６０】サブシステム内のあるディスク制御装置に障害が発生した場合は、正常なディスク制御装置のキャッシュに保持している、該障害発生ディスク制御装置に接続したドライブのデータは無効化するようにしたので、障害時にも障害が伝播することはない。

【００６１】ディスク制御装置間の通信手段は、ホストコンピュータと接続が可能なチャネルの一部と、該チャネル同士を接続するスイッチであるようにしたので、専用のディスク制御装置間接続手段を持たないディスク制御装置からなるサブシステムにおいても、複数のディスク制御装置間でキャッシュアクセスが可能となる。

【００６２】ディスク制御装置内に備えたキャッシュメモリを制御するための制御情報として、チャネルとディスク制御装置と論理ディスク毎のアクセス頻度を保持するアクセスログテーブルを設け、ある論理ディスクへのアクセスを受領するチャネルのうち、アクセス頻度が最も高いチャネルと該アクセス先の論理ディスクが同一のディスク制御装置に接続されているかを判定し、同一でない場合は、該論理ディスクを該アクセス頻度が最も高いチャネルが接続されたディスク制御装置のドライブ上に再配置するようにした。また、同一である場合は、該論理ディスクにアクセスする他のチャネルを使用するホストコンピュータは、該論理ディスクを接続するディスク制御装置のチャネルを使用するようにした。その結果、サブシステム内のデータ配置の最適化を図ることができ、ディスク制御装置間パスの使用頻度を低く抑えることが可能となり、ディスク制御装置間パスに要求される帯域を低く抑えられるので低コスト化できる。

【図面の簡単な説明】

【図１】本発明に係るディスク制御装置の概要を示すブロック図の一例である。

【図２】本発明に係るディスク制御装置のキャッシュ制御情報を示すブロック図の一例である。

【図３】本発明に係る動作全体の一例を示す流れ図である。

【図４】本発明に係る更新アクセス要求処理の一例を示す流れ図である。

【図５】本発明に係るコヒーレンス処理の一例を示す流れ図である。

【図６】本発明に係るコヒーレンス処理の一例を示す流れ図である。

【図７】本発明に係る参照アクセス要求処理の一例を示す流れ図である。

【図８】本発明に係る参照アクセス要求処理の一例を示す流れ図である。

【図９】本発明に係るキャッシュ管理方法の一例を示す流れ図である。

【図１０】本発明に係る参照アクセス要求処理の一例を示す流れ図である。

【図１１】本発明に係るディスク制御装置のキャッシュを示すブロック図の一例である。

【図１２】本発明に係るディスク制御装置のキャッシュを示すブロック図の一例である。

【図１３】本発明に係るキャッシュ管理方法の一例を示す流れ図である。

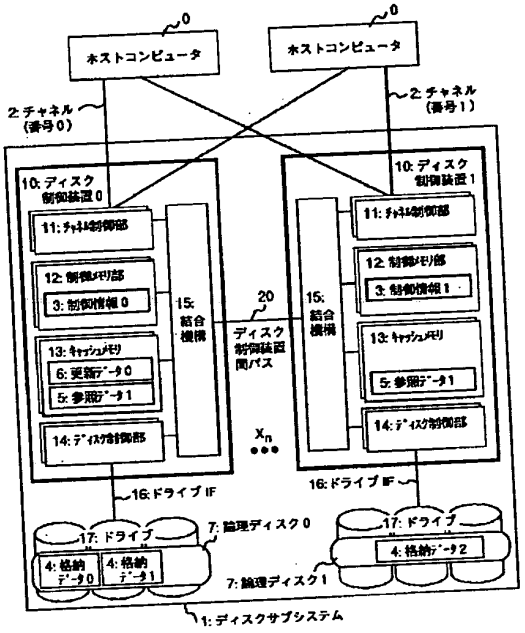
【図１４】本発明に係るディスク制御装置の概要を示すブロック図の他の一例である。

【図１５】本発明に係るディスク制御装置の概要を示すブロック図の他の一例である。
【図１６】本発明に係るディスク制御装置のデータ配置方法の一例を示す流れ図である。
【図１７】本発明に係る従来のディスク制御装置の概要を示すブロック図である。
【図１８】本発明に係る従来のディスク制御装置の概要を示すブロック図である。
【符号の説明】

０・・・ホストコンピュータ、１・・・ディスクサブシステム、２・・・チャネル、３・・・制御情報、４・・・格納データ、５・・・参照データ、６・・・更新データ、７・・・論理ディスク、１０・・・ディスク制御装置、１１・・・チャネル制御部、１２・・・制御メモリ、１３・・・キャッシュメモリ部、１４・・・ディスク制御部、１５・・・結合機構、１６・・・ドライブＩＦ、１７・・・ドライブ、２０・・・ディスク制御装置間パス。

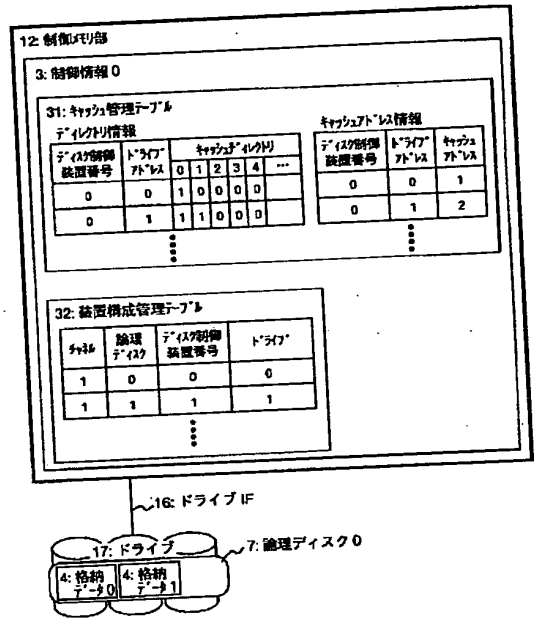
【図１】

図　1



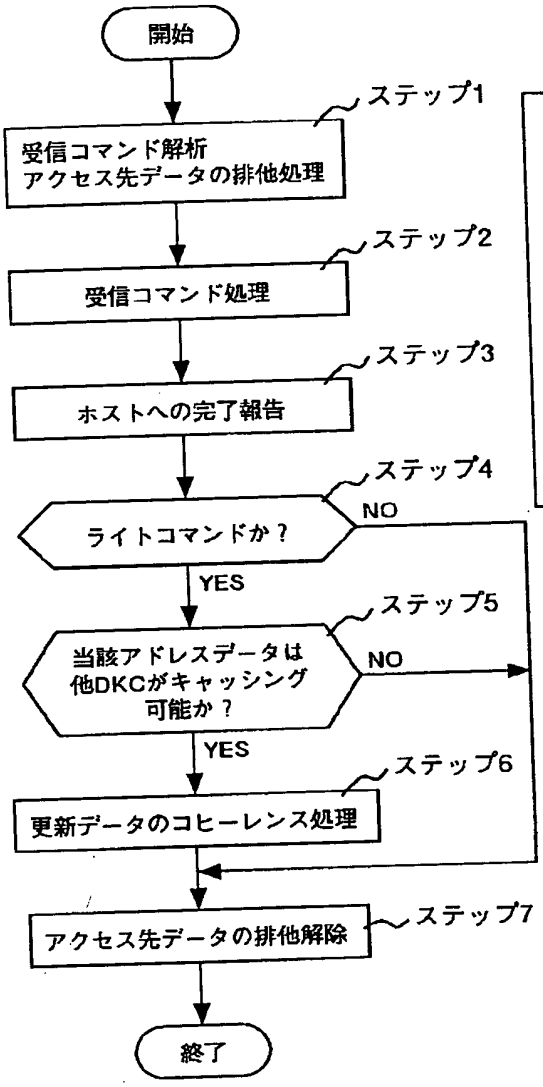
【図２】

図　2



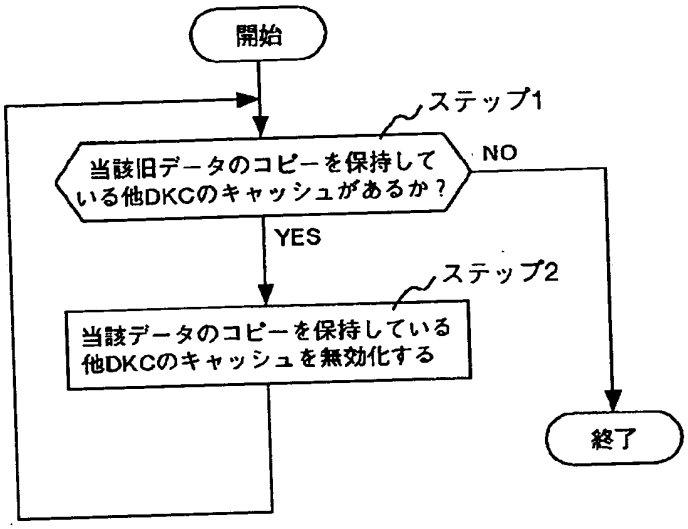
【図３】

図　3



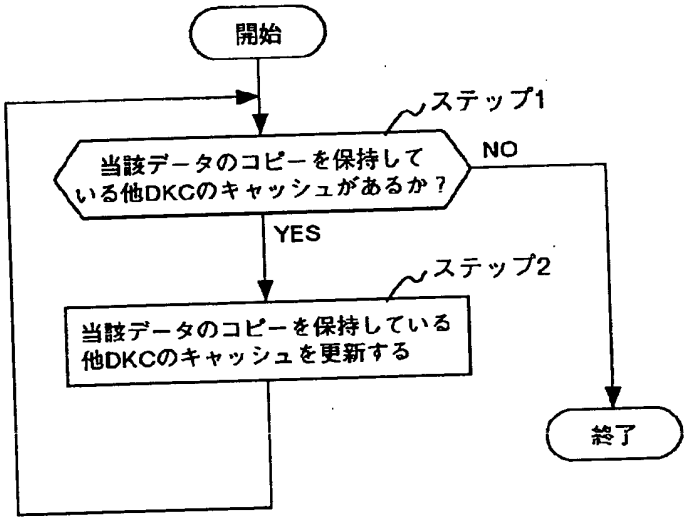
【図５】

図　5



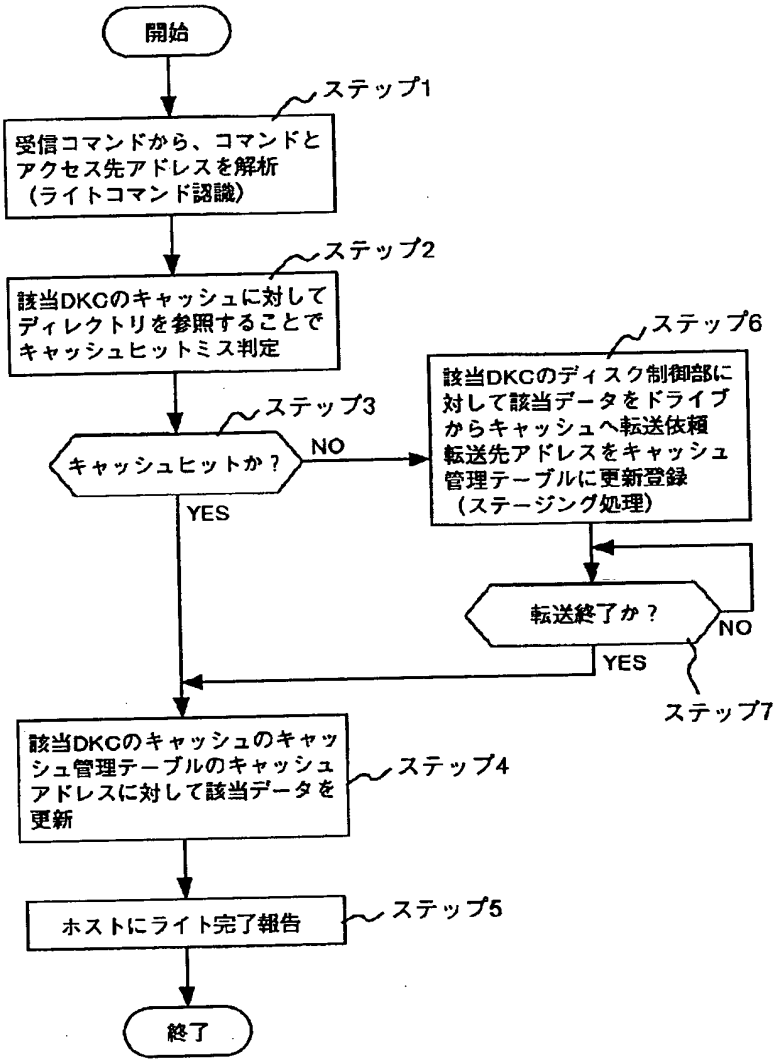
【図６】

図　6



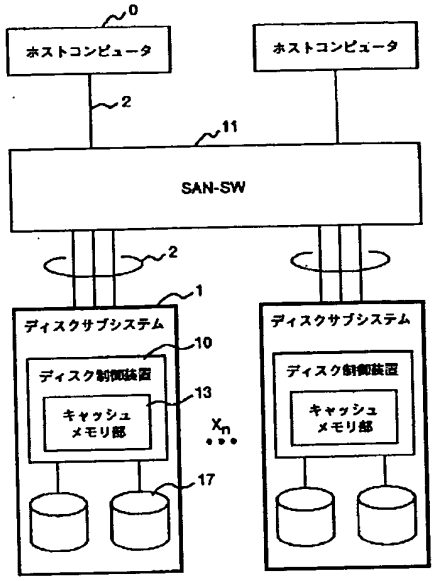
【図４】

図　4


開始
ステップ1
受信コマンドから、コマンドとアクセス先アドレスを解析（ライトコマンド認識）
ステップ2
該当DKCのキャッシュに対してディレクトリを参照することでキャッシュヒットミス判定
ステップ3
キャッシュヒットか？
NO
YES
ステップ6
該当DKCのディスク制御部に対して該当データをドライブからキャッシュへ転送依頼
転送先アドレスをキャッシュ管理テーブルに更新登録（ステージング処理）
転送終了か？
NO
YES
ステップ7
該当DKCのキャッシュのキャッシュ管理テーブルのキャッシュアドレスに対して該当データを更新
ステップ4
ホストにライト完了報告
ステップ5
終了

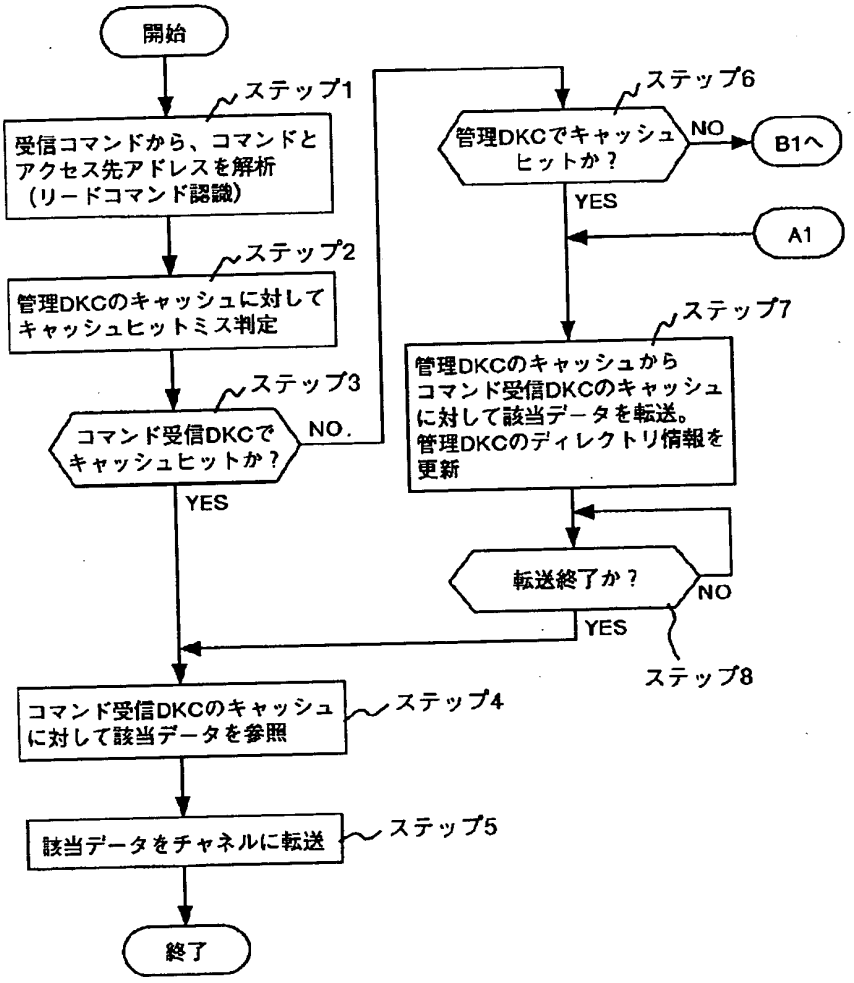
【図１８】

図　１８


0
ホストコンピュータ
ホストコンピュータ
2
11
SAN-SW
2
1
ディスクサブシステム
10
ディスク制御装置
13
キャッシュメモリ部
Xn
17
ディスクサブシステム
ディスク制御装置
キャッシュメモリ部

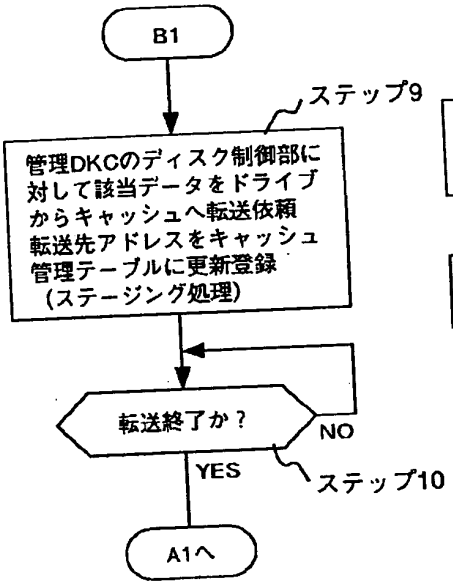
【図７】

図　7


開始
ステップ1
受信コマンドから、コマンドとアクセス先アドレスを解析（リードコマンド認識）
ステップ2
管理DKCのキャッシュに対してキャッシュヒットミス判定
ステップ3
コマンド受信DKCでキャッシュヒットか？
NO
YES
ステップ4
コマンド受信DKCのキャッシュに対して該当データを参照
ステップ5
該当データをチャネルに転送
終了
ステップ6
管理DKCでキャッシュヒットか？
NO
B1へ
YES
A1
ステップ7
管理DKCのキャッシュからコマンド受信DKCのキャッシュに対して該当データを転送。管理DKCのディレクトリ情報を更新
転送終了か？
NO
YES
ステップ8

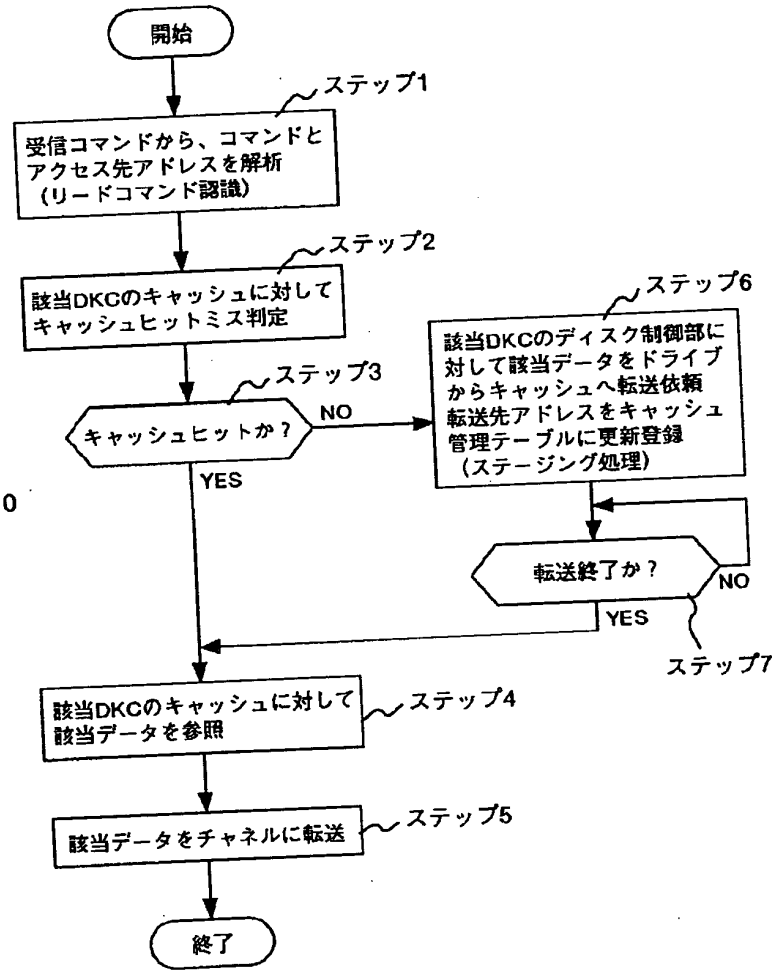
【図８】

図　８


B1
ステップ9
管理DKCのディスク制御部に
対して該当データをドライブ
からキャッシュへ転送依頼
転送先アドレスをキャッシュ
管理テーブルに更新登録
（ステージング処理）
転送終了か？
NO
YES
ステップ10
A1へ

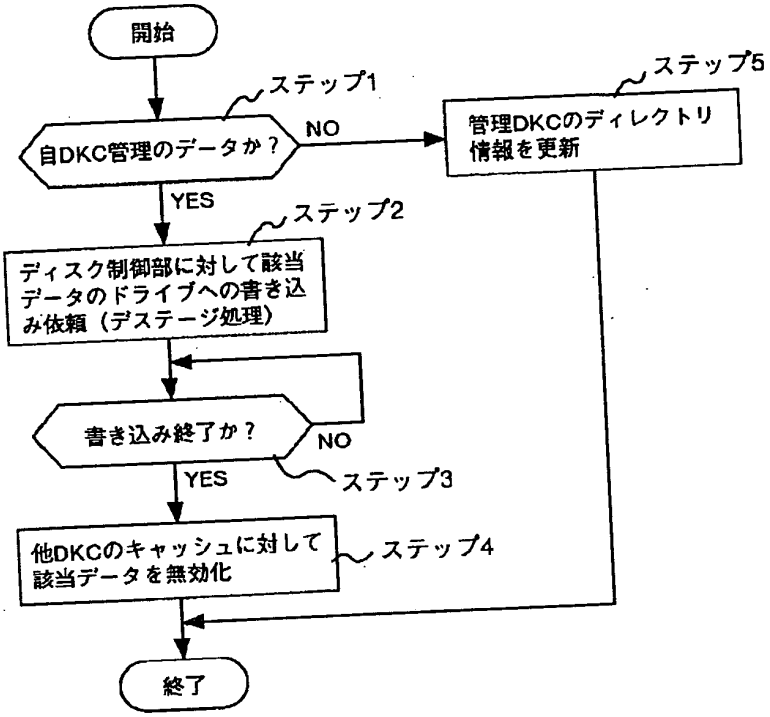
【図１０】

図　１０


開始
ステップ1
受信コマンドから、コマンドと
アクセス先アドレスを解析
（リードコマンド認識）
ステップ2
該当DKCのキャッシュに対して
キャッシュヒットミス判定
ステップ3
キャッシュヒットか？
NO
YES
ステップ6
該当DKCのディスク制御部に
対して該当データをドライブ
からキャッシュへ転送依頼
転送先アドレスをキャッシュ
管理テーブルに更新登録
（ステージング処理）
転送終了か？
NO
YES
ステップ7
該当DKCのキャッシュに対して
該当データを参照
ステップ4
該当データをチャネルに転送
ステップ5
終了

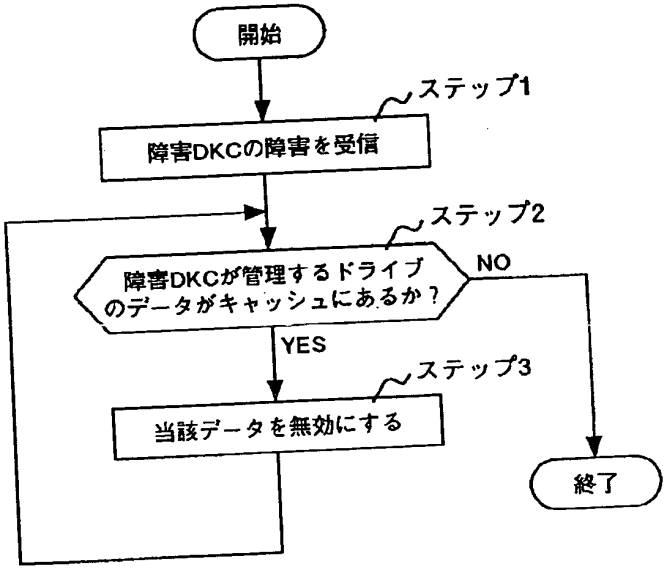
【図９】

図　９



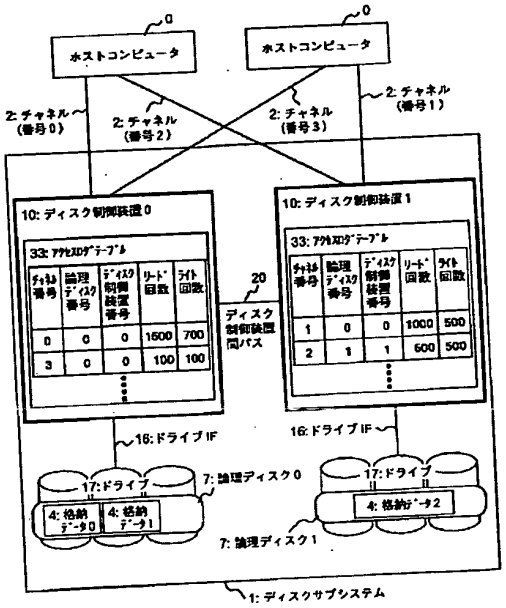
【図１３】

図　１３



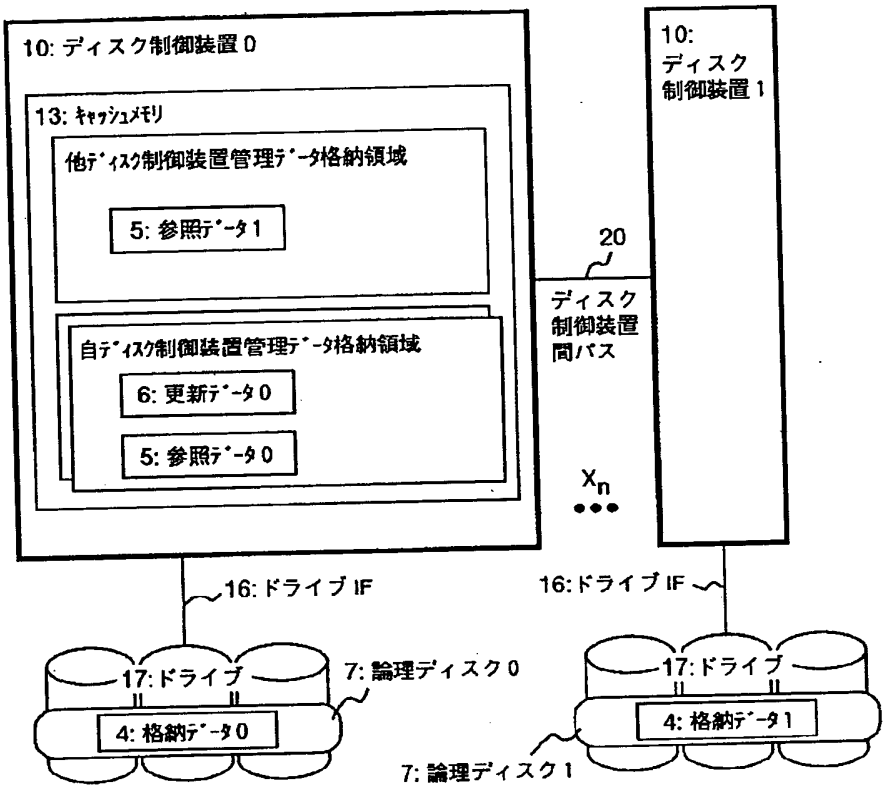
【図１５】

図　１５



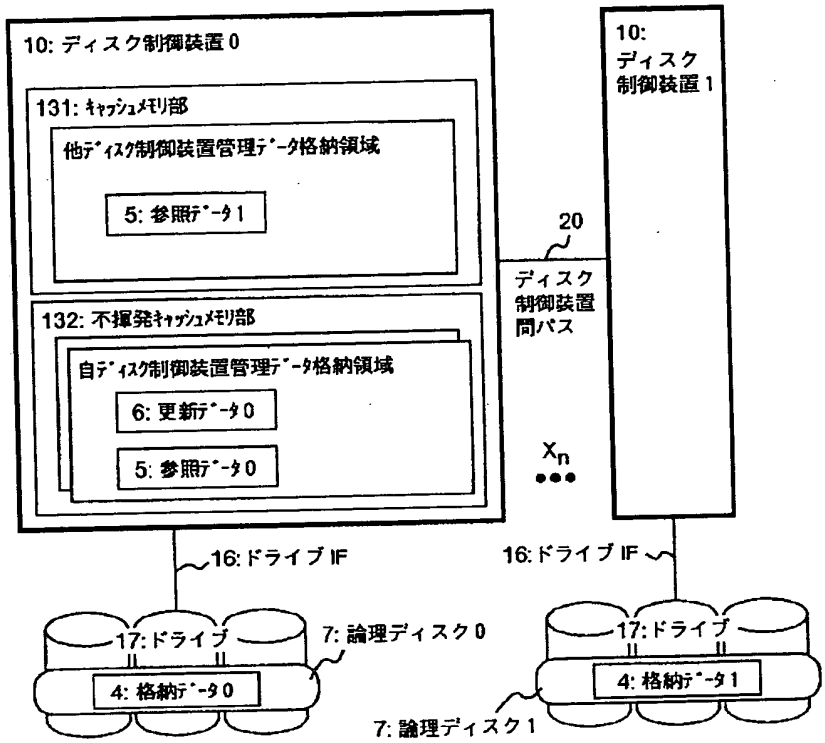
【図１１】

図　１１


10: ディスク制御装置０
13: ｷｬｯｼｭﾒﾓﾘ
他ﾃﾞｨｽｸ制御装置管理ﾃﾞｰﾀ格納領域
5: 参照ﾃﾞｰﾀ１
自ﾃﾞｨｽｸ制御装置管理ﾃﾞｰﾀ格納領域
6: 更新ﾃﾞｰﾀ０
5: 参照ﾃﾞｰﾀ０
10:
ディスク
制御装置１
20
ディスク
制御装置
間パス
Xn
16: ドライブ IF
16: ドライブ IF
17: ドライブ
4: 格納ﾃﾞｰﾀ０
7: 論理ディスク０
17: ドライブ
4: 格納ﾃﾞｰﾀ１
7: 論理ディスク１

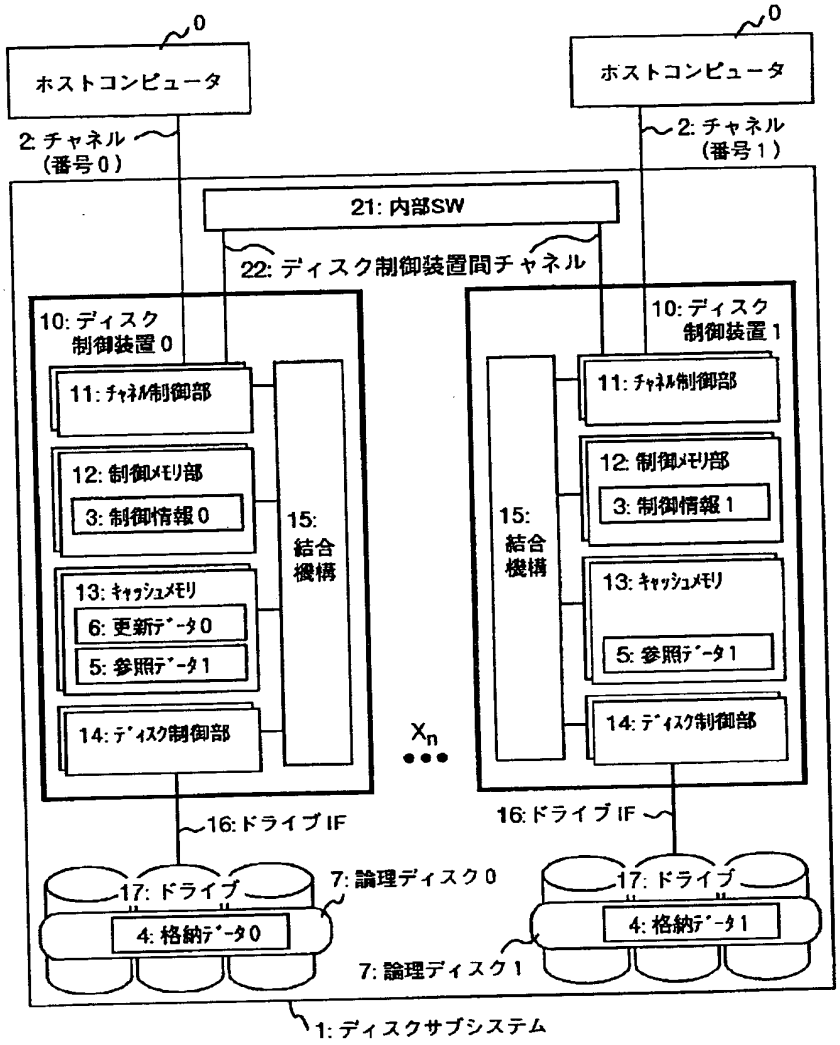
【図１２】

図　１２


10: ディスク制御装置 0
131: ｷｬｯｼｭﾒﾓﾘ部
他ﾃﾞｨｽｸ制御装置管理ﾃﾞｰﾀ格納領域
5: 参照ﾃﾞｰﾀ 1
132: 不揮発ｷｬｯｼｭﾒﾓﾘ部
自ﾃﾞｨｽｸ制御装置管理ﾃﾞｰﾀ格納領域
6: 更新ﾃﾞｰﾀ 0
5: 参照ﾃﾞｰﾀ 0
20
ディスク
制御装置
間パス
10:
ディスク
制御装置 1
Xn
16: ドライブ IF
16: ドライブ IF
17: ドライブ
7: 論理ディスク 0
4: 格納ﾃﾞｰﾀ 0
17: ドライブ
4: 格納ﾃﾞｰﾀ 1
7: 論理ディスク 1

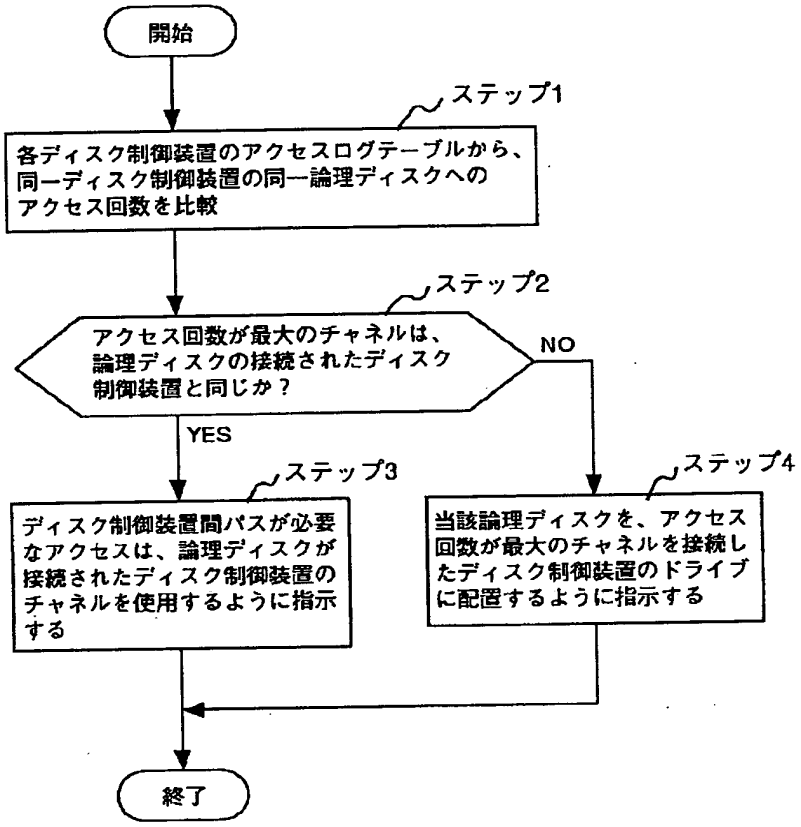
【図１４】

図　１４


0
ホストコンピュータ
0
ホストコンピュータ
2: チャネル
(番号０)
2: チャネル
(番号１)
21: 内部SW
22: ディスク制御装置間チャネル
10: ディスク
制御装置０
11: ﾁｬﾈﾙ制御部
12: 制御ﾒﾓﾘ部
3: 制御情報０
13: ｷｬｯｼｭﾒﾓﾘ
6: 更新ﾃﾞｰﾀ０
5: 参照ﾃﾞｰﾀ１
14: ﾃﾞｨｽｸ制御部
15:
結合
機構
10: ディスク
制御装置１
11: ﾁｬﾈﾙ制御部
12: 制御ﾒﾓﾘ部
3: 制御情報１
15:
結合
機構
13: ｷｬｯｼｭﾒﾓﾘ
5: 参照ﾃﾞｰﾀ１
14: ﾃﾞｨｽｸ制御部
Xn
16: ドライブIF
16: ドライブIF
17: ドライブ
7: 論理ディスク０
4: 格納ﾃﾞｰﾀ０
17: ドライブ
4: 格納ﾃﾞｰﾀ１
7: 論理ディスク１
1: ディスクサブシステム

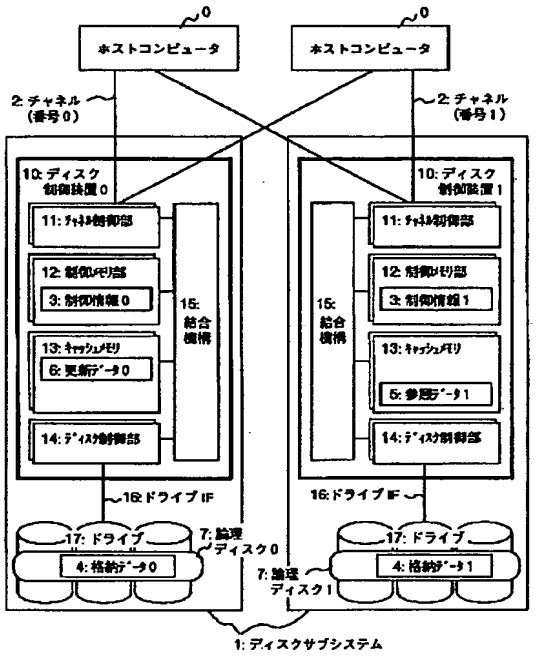
【図１６】

図　１６


開始
ステップ1
各ディスク制御装置のアクセスログテーブルから、同一ディスク制御装置の同一論理ディスクへのアクセス回数を比較
ステップ2
アクセス回数が最大のチャネルは、論理ディスクの接続されたディスク制御装置と同じか？
NO
YES
ステップ3
ディスク制御装置間パスが必要なアクセスは、論理ディスクが接続されたディスク制御装置のチャネルを使用するように指示する
ステップ4
当該論理ディスクを、アクセス回数が最大のチャネルを接続したディスク制御装置のドライブに配置するように指示する
終了

【図１７】

図　１７



フロントページの続き

(51)Int.Cl.⁷		識別記号	FI		テーマコード(参考)
Ｇ０６Ｆ	12/08	５５７	Ｇ０６Ｆ	12/08	５５７
	13/00	３０１		13/00	３０１Ｐ

(72)発明者　藤林　昭
東京都国分寺市東恋ヶ窪一丁目280番地
株式会社日立製作所中央研究所内

Ｆターム(参考)　5B005 JJ01 KK14 MM12 PP11 PP21
WW11
5B014 EA04 EB05 FB07
5B065 BA01 CA07 CA11 CC08 CE12
EA18 EA25 EA31
5B083 AA08 BB01 CC04 CD13 DD08
EE08 EF11 GG04